

IOLAN DS Family

DS1/TS2 User's Guide

Version 1.9

Part #5500162-19

February 2007

Copyright Statement

This document must not be reproduced in any way whatsoever, either printed or electronically, without the consent of:

Perle Systems Limited,
60 Renfrew Drive
Markham, ON
Canada
L3R 0E1

Perle reserves the right to make changes without further notice, to any products to improve reliability, function, or design.

Perle, the Perle logo, and IOLAN are trademarks of Perle Systems Limited.

Microsoft, Windows 98, Windows NT, Windows 2000, Windows Server 2003, Windows XP, and Internet Explorer are trademarks of Microsoft Corporation.

Netscape is a trademark of Netscape Communications Corporation.

Mozilla Firefox is a trademark of the Mozilla Foundation.

Solaris is a registered trademark of Sun Microsystems, Inc. in the USA and other countries.

Perle Systems Limited, 2004-2006.

FCC Note The IOLAN Device Server has been found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions in this Guide, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at his/her own expense.

EN 55022: 1998, Class A, Note

WARNING This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.



Caution: The IOLAN Device Server is approved for commercial use only.



Table of Contents

Preface	19
About This Book	19
Intended Audience	19
Documentation	19
Typeface Conventions	20
Online Help	20
Contacting Technical Support	21
Making a Technical Support Query	21
Who To Contact	21
Have Your Product Information Ready	21
Making a support query via the Perle web page	21
Repair Procedure	22
Feedback on this Manual	22
Chapter 1 Introduction	23
About the IOLAN Device Server	23
IOLAN Device Server Models	23
Device Server Features	24
Hardware	24
Software	24
Security	25
Supported Products/Versions	25
Web Browsers	25

Typical Applications Summary	25
Managing the Device Server	25
Managing/Accessing devices attached to the Device Server.....	25
Network Security.....	26
Chapter 2 Installation	27
Introduction.....	27
IOLAN Device Server Components.....	27
What's Included	27
What You Need to Supply	27
Available Accessories	27
Getting to Know Your Device Server	28
DS1	28
TS2	28
LED Guide.....	29
Console Mode vs. Serial Mode	29
Powering Up the Device Server	29
Power Supply Requirements	29
Serial Only Models	29
I/O Models	30
Serial Only Models.....	30
I/O Models.....	30
Setting Jumpers	31
1-Port Device Server.....	31
Device Server DB25 Male/Female	31
Device Server RJ45.....	31
Device Server DB9	32
2-Port Device Server.....	32
Digital I/O Module.....	33
Analog Input Module	34

Wiring I/O Diagrams.....	34
Digital I/O.....	34
Digital Input Wet Contact	34
Digital Input Dry Contact	35
Digital Output Sink	35
Digital Output Source	35
Analog Input.....	36
Current	36
Voltage	36
Temperature Input.....	36
Thermocouple	36
RTD 2-Wire	37
RTD 3-Wire	37
RTD 4-Wire	37
Relay Output	37
Normally Open Contact.....	37
Normally Closed Contact	38
Setting an Initial IP Address	38
Using DeviceManager	39
Using a Direct Connection.....	40
Using DHCP/BOOTP.....	40
Using ARP-Ping	41
IPv6 Network.....	41
Pinouts.....	42
DB25 Male	42
DB25 Female	43
RJ45.....	44
DB9 Male (Serial Only)	44
DB9 Male I/O.....	45
EIA-232 Cabling Diagrams	46
Terminal DB25 Connector	46
DB25 Male	46
DB25 Female	46
RJ45.....	47
DB9 Male	47
Modem DB25 Connector.....	48
DB25 Male	48
RJ45.....	48
DB9 Male	49

Chapter 3 Configuration Methods	51
Introduction.....	51
DeviceManager	51
WebManager	52
Using the WebManager	52
CLI.....	52
Menu	52
Accessing the Menu	52
Menu Conventions.....	53
DHCP/BOOTP.....	53
SNMP	54
Chapter 4 Configuring the Device Server	55
Introduction.....	55
Configuring the Device Server	55
General Device Server Configuration	55
Device Server Services	55
TruePort.....	56
Hardware Configuration	56
Ethernet Connection.....	56
Serial Connection	56
Other.....	56
Modbus Configuration	57
Overview	57
Configuring a Master Gateway	57
Configuring a Slave Gateway	57
Modbus Gateway Settings	58
Modbus Master Gateway.....	58
Modbus Slave Gateway.....	58
Modbus Line Settings.....	59
Modbus Master Settings.....	59
Modbus Slave Settings.....	59
Example Scenario	60

Machine To Machine Connections	62
Users Connecting to Serial Devices	62
Users Connecting to the LAN	62
Connecting To the Device Server	63
Connecting Through the Device Server	63
Setting Up Lines.....	63
DSLogin.....	63
Direct/Silent/Reverse Connections.....	64
Virtual Modems.....	64
BIDIR.....	64
TruePort.....	64
Signal I/O	64
UDP	65
Serial Tunnel Settings.....	66
Setting Up Users	66
User Accounts	66
User Levels	67
Sessions.....	67
Users From LAN to Device Server to Serial Device	67
Easy Port Access Menu	67
Reverse Sessions and Multisessions.....	68
Configuring Network Options	69
Hosts.....	69
Gateways.....	69
Syslog.....	69
SNMP	69
Configuring Time	69
Language support.....	69
Loading a Supplied Language	70
Translation Guidance.....	70
Software Upgrades and Language Files	71

Downloading Terminal Definitions	71
Creating Terminal Definition Files.....	71
TFTP Configuration	73
Resetting Configuration Parameters	73
Lost Admin Password	73
DHCP/BOOTP	73
DHCP/BOOTP Parameters	74
Creating Custom Applications	74
I/O Model Features	74
Failsafe Timer.....	74
Alarms.....	75
UDP	75
UDP Unicast Format.....	75
UDP Unicast Example	76
I/O Modbus Slave	76
Modbus Serial Application Connected to the Serial Port.....	76
Modbus Serial Application Connected to the Network	77
Modbus TCP Application	77
Modbus I/O Access.....	77
Function Codes	77
I/O Coil/Register Descriptions	78
Serial Port Coil/Register Descriptions	79
A4/T4 Registers.....	79
A4D2/A4R2 Registers	80
D4/D2R2 Registers.....	81
Serial Signals.....	81
TruePort.....	82
TruePort/Modbus Combination.....	82
API Over TruePort Only.....	82
Digital Channels.....	83
Digital Input.....	83
Digital Output.....	84
Temperature Channels	85
Analog Channels.....	86
Relay Channels	87
Serial Signals	87

SNMP Traps	88
Calibrating Analog Input.....	89
Calibrating Voltage.....	89
Calibrating Current.....	89
Calibrating Temperature Input	89
Calibrating Thermocouple	89
Calibrating RTD.....	89
Chapter 5 Using the DeviceManager	91
Introduction	91
Starting a New Session	91
Managing a Device Server	92
Populating the Device Server List	92
Assigning a Temporary IP Address to a New Device Server.....	93
Adding/Deleting Static Device Servers	93
Creating a New Device Server Configuration	94
Opening an Existing Configuration File	94
Connecting to a Device Server	94
Managing a Device Server	95
DeviceManager Work Flow.....	95
Creating/Editing Configuration Files	95
Working With the Device Server Configuration.....	95
Working With a Local Configuration File	95
Configuring the Server	96
Configuring the Main Server Window.....	96
Server.....	96
Services	96
Configuring Advanced Server Settings.....	97
Configuring TruePort Baud	98
Configuring the Hardware	98
Custom App/Plugin	98

Configuring Lines	99
Advanced Line Settings	101
Service Settings	103
DSLogin.....	103
Raw Settings	104
Telnet Settings.....	104
BIDIR Settings.....	105
UDP Settings.....	106
VModem Settings	107
Server Tunnel Settings.....	108
Client Tunnel Settings	108
Modbus Slave Settings.....	109
Modbus Master Settings.....	110
Remote IP Slave Mappings.....	111
Custom App/Plugin Settings.....	112
TruePort Settings.....	113
Packet Forwarding.....	114
Configuring Modems	116
Configuring I/O	117
Global Settings.....	117
Temperature Settings.....	117
Failsafe Timer Settings.....	117
Modbus Settings.....	118
TruePort Settings.....	118
UDP Settings	118
Channels.....	120
Digital Output.....	120
Digital Input.....	121
Relay	123
Analog	124
Basic Alarm Settings	125
Advanced Alarm Settings	126
Temperature	127
Configuring Users	127
Configuring Line Access.....	129
Configuring Sessions.....	129
Configuring the Default User	130

Configuring the Network	130
Configuring Hosts	130
Configuring SNMP	131
Configuring TFTP	132
Configuring Gateways	132
Configuring Syslog	133
Configuring Administration Tasks	134
Configuring Bootup Files	134
Configuring the MOTD File	134
I/O Status/Control	135
Statistics	135
Tools	136
Saving a Configuration To File.....	136
Getting a Configuration File	136
Configuring Multiple Device Servers.....	136
Downloading Device Server Firmware	137
Setting the Device Server's Date and Time.....	138
Rebooting the Device Server.....	138
Resetting the Device Server to Factory Defaults	138
Resetting a Line.....	138
Custom Files	139
Saving Crashes to a Dump File	139
Downloading Terminal Definitions.....	139
Downloading a Language File.....	139
Downloading a Custom App/Plugin File.....	139
I/O Channels.....	139
Calibrating Analog Channels.....	139
Resetting Calibration Data	140
Setting DeviceManager Options	141

Chapter 6 Command Line Interface	143
Introduction.....	143
CLI Conventions	143
Command Syntax.....	143
Command Shortcuts.....	144
Command Options.....	144
Server Commands	145
Server Commands	145
Set Custom-App	145
Set Server.....	145
Set Service	147
Show Custom-App.....	147
Show Modbus.....	147
Show Server	147
Hardware Commands	148
Set Ethernet.....	148
Show Hardware	148
Modbus Commands.....	148
Set Modbus Gateway	148
Show Modbus.....	149
TruePort Baud Commands	150
Set TruePort Remap-Baud	150
Show TruePort.....	150
User Commands	150
Logged Into the Device Server Commands.....	150
Admin	150
Help	150
Kill Line	150
Kill Session	150
Logout.....	151
Menu.....	151
Ping	151
Resume	151
Screen	151
Set Termttype	152
Set User.....	152
Set User Session.....	153
Show Line Users	153
Syslog Console.....	153
Show Sessions	153

Show Termttype	153
Start.....	154
Telnet	154
Version	155
Configuring Users	155
Add User	155
Delete User	155
Set Default User	156
Set User	158
Set User Session	160
Show Default User	160
Show User.....	160
Line Commands	161
Line Commands.....	161
Set Line	161
Set Line Interface	164
Set Line Service	165
Set Modem.....	167
Set Termttype.....	167
Show Line	167
Line Service Commands.....	168
Set Custom-App.....	168
Set Telnet-Client	168
Set UDP	169
Set Vmodem	169
Set Modbus-Slave Line	170
Set Modbus-Master Line	171
Show Custom-App	171
Show Interface	172
Show Modbus	172
Show Telnet-Client.....	172
Show UDP.....	172
Show Vmodem.....	172
Modem Commands	172
Add Modem.....	172
Delete Modem.....	172
Show Modems	172
Packet Forwarding Commands.....	173
Set Packet-Forwarding Line.....	173
Show Packet-Forwarding Line	175

Network Commands	175
SNMP Commands	175
Add Community	175
Add Trap	175
Delete Community	176
Delete Trap	176
Set SNMP	176
Show SNMP	176
TFTP Commands	177
Set Server TFTP	177
Hosts Commands	177
Add Host	177
Delete Host	177
Set Host	177
Show Hosts	177
Gateway Commands	178
Add Gateway	178
Delete Gateway	178
Set Gateway	179
Show Gateways	179
Logging Commands	180
Set Syslog	180
Show Syslog	180
Time Commands	180
Set Time	180
Show Time	180
Time/Date Setting Commands	181
Set Date	181
Set Time	181
Show Date	181
Show Time	181
Administration Commands	181
Bootup Commands	181
Reboot	181
Reset	181
Reset Factory	181
Save	181
Set Bootup	182
Show ARP	182
Show Bootup	182

TFTP File Transfer Commands	182
Netload.....	182
Netsave	183
MOTD Commands	183
Set MOTD	183
Show MOTD.....	184
Statistic Commands	184
Configuration Statistics	184
Show Netstat.....	184
Show Modbus Statistics	184
Show Netstat Statistics	184
Show Routes.....	184
Run-Time Statistics	185
Delete Arp	185
Show Arp.....	185
Show Serial	185
Uptime.....	185
I/O Commands.....	185
Global I/O Commands	185
Set IO UDP	185
Set IO Failsafe	186
Set IO Modbus	186
Set IO Temperature-Scale	186
Set Line.....	187
Set Line Service	187
Set IOChannel	187
Set IOChannel Mode.....	187
Set IOChannel Digital I/O.....	187
Set IOChannel Digital Input.....	188
Set IOChannel Digital Input (Serial Pins)	189
Set IOChannel Digital Output.....	190
Set IOChannel Digital Output (Serial Pins)	191
Set IOChannel Relay	192
Set IOChannel Analog (True Analog)	193
Set IOChannel Analog (Temperature)	195
Kill IOChannel	197
Show IO	197
Show IOChannel	197

I/O Channel Control Commands.....	198
Digital Output.....	198
Digital Input.....	198
Relay	198
Analog Input	198
Calibrating Analog Input (Analog/Temperature).....	199
Calibrate Analog	199
Reset Calibration	199
Appendix A Troubleshooting	201
Introduction.....	201
Hardware Problems	201
Communication Issues	201
DeviceManager Problems.....	202
Host Problems	202
Login Problems	202
Problems with Terminals	203
Unknown IP Address.....	203
DHCP/BOOTP Problems	204
Language Problems	204
Long Reboot Cycle.....	204
I/O Models	204
Appendix B Utilities	205
Introduction.....	205
TruePort.....	205
API I/O Access Over TruePort	206
API Request Format.....	206
API Response Format.....	206
Error Codes	207

Appendix C Accessories	209
Introduction	209
Starter Kit (Adapters/Cable).....	209
RJ45F to DB25M DTE Crossover Adapter.....	210
RJ45F to DB25M DCE Modem Adapter	211
RJ45F to DB25F DTE Crossover Adapter	212
RJ45F to DB9M DTE Crossover Adapter.....	213
RJ45F to DB9F DTE Crossover Adapter	214
Sun/Cisco RJ45M Connector Cable for Rack Mount Models	214
Glossary	215
Index	217



Preface

About This Book

This guide provides the information you need to:

- configure the Device Server
- incorporate the Device Server into your production environment

Intended Audience

This guide is for administrators who will be configuring the Device Server.

Some prerequisite knowledge is needed to understand the concepts and examples in this guide:

- If you are using an external authentication application(s), working knowledge of the authentication application(s).
- Knowledge of TFTP, the transfer protocol the Device Server uses.

Documentation

The following documentation is included on the Device Server installation CD:

- *IOLAN DS1/SDS1 Device Server Quick Start Guide*
- *IOLAN Device Server User's Guide*
- *TruePort User Guide*
- *TruePort Installation and Configuration Guide for Windows NT*
- Online Help in the DeviceManager (automatically installed with the DeviceManager application)
- Link to knowledge base

Typeface Conventions

Most text is presented in the typeface used in this paragraph. Other typefaces are used to help you identify certain types of information. The other typefaces are:

Typeface Example	Usage
At the C: prompt, type: add host	This typeface is used for code examples and system-generated output. It can represent a line you type in, or a piece of your code, or an example of output.
Set the value to TRUE .	The typeface used for TRUE is also used when referring to an actual value or identifier that you should use or that is used in a code example.
subscribe <i>project subject</i> run <i>yourcode</i> .exec	The italicized portion of these examples shows the typeface used for variables that are placeholders for values you specify. This is found in regular text and in code examples as shown. Instead of entering <i>project</i> , you enter your own value, such as <code>stock_trader</code> , and for <i>yourcode</i> , enter the name of your program.
File, Save	This typeface and comma indicates a path you should follow through the menus. In this example, you select Save from the File menu.
<i>IOLAN DS Family User's Guide</i>	This typeface indicates a book or document title.
See Chapter 1, Introduction on page 23 for more information.	This indicates a cross-reference to another chapter or section that you can click on to jump to that section.

Online Help

Online help is provided in the DeviceManager. You can click on the What's This button ( or ) and then click on a field to get field-level help. Or, you can press the **F1** key to get window-level help. You can also get the *User's Guide* online by selecting **Help, Help Topics**.

Contacting Technical Support

Making a Technical Support Query

Who To Contact

Note: Perle offers free technical support to Perle Authorised Distributors and Registered Perle Resellers.

If you bought your product from a registered Perle supplier, you must contact their Technical Support department; they are qualified to deal with your problem.

Have Your Product Information Ready

When you make a technical support enquiry please have the following information ready:

Item	Write Details Here
Product Name	
Problem Description	
Your Name	
Company Name and Address	
Country	
Phone Number	
Fax Number	
Email Address	

Making a support query via the Perle web page

If you have an internet connection, please send details of your problem to Technical Support using the email links provided on the Perle web site in the **Support** area.

Click here to access our website at the following URL:

<http://www.perle.com>

Repair Procedure

Before sending a Device Server for repair, you must contact your Perle supplier. If, however, you bought your product directly from Perle you can contact directly.

Customers who are in Europe, Africa or Middle East can submit repair details via a website form. This form is on the Perle website, www.perle.com, in the **Support** area.

Click here to access our web site at the following URL:

http://www.perle.com/support_services/rma_form.asp

Feedback on this Manual

If you have any comments or suggestions for improving this manual please email Perle using the following address;

Email: ptac@perle.com

Please include the **title**, **part number** and **date** of the manual (you can find these on the title page at the front of this manual).



Introduction

About the IOLAN Device Server

The Device Server is an ethernet communications/terminal server that allows serial devices to be connected directly to LANs. The Device Server can connect to a wide range of devices including:

- Terminals for multi-user UNIX systems
- Data acquisition equipment (manufacturing, laboratory, scanners, etc.)
- Retail point-of-sale equipment (bar coding, registers, etc.)
- PCs using terminal emulation
- Modems for remote access and Internet access
- ISDN adapters for branch remote access and Internet access
- All types of serial printers

The performance and flexibility of the Device Server allows you to use a wide range of high speed devices in complex application environments. The Device Server will work in any server environment running TCP/UDP/IP.

IOLAN Device Server Models

The IOLAN Device Server comes in several different models to meet your production environment needs:

- **DS**—Offered as a 1-port unit, this model provides basic Device Server functionality. This model can be ordered with RJ45, DB9 male, DB25 female, or DB25 male connection options. There is also a line of DS models that support Analog Input, Temperature Input, Relay Output, and/or Digital I/O.
- **TS**—The model does everything the DS model does plus has two RJ45 serial ports (supports EIA-232 only). This model does not support the power out/power in pins or I/O.
- **SDS**—This model does everything the DS model does plus additional features such as external authentication, SSH, SSL, port buffering, email alerts, RIP, DNS/WINS, plus much more. This model has an EIA-232/422/485 switchable interface. Rack mount models have a dedicated Console port and support gigabit Ethernet. The 1-port model can be ordered with RJ45, DB9 male, DB25 female, or DB25 male connection options. Some SDS models support Power Over Ethernet or have an internal modem. There is also a line of SDS models that support Analog Input, Temperature Input, Relay Output, and/or Digital I/O; all models in this line are extended temperature models, meaning that they can operate in higher temperature environments.
- **STS**—This model does everything the DS model does plus additional features such as external authentication, SSH, SSL, port buffering, email alerts, RIP, DNS/WINS, plus much more. This model has an EIA-232 interface. Rack mount models have a dedicated Console port and support gigabit Ethernet. Some models support dual input DC power.

- **SCS**—This model does everything the DS model does plus additional features such as external authentication, SSH, SSL, port buffering, email alerts, RIP, DNS/WINS, plus much more. This model has an EIA-232 interface. Rack mount models have a dedicated Console port. This model comes equipped with PCI interface (supports the Perle PCI modem card), gigabit support, dual Ethernet, and can have dual AC power.

Device Server Features

The Device Server is a communications server with 1 port for making serial network connections. It attaches to your TCP/IP network and allows serial devices such as modems, terminals, or printers to access the LAN. It also allows LAN attached devices to access serial devices attached to the Device Server.

Hardware

The Device Server hardware features include:

- Auto sensing 10/100 RJ45 Ethernet interface.
- Universal, software-selectable EIA-232/422/485 interface.
- Full modem control using DTR, DSR, CTS, RTS and DCD.
- Tx and Rx activity indicators.
- External AC power supply or power over serial.
- LEDs for diagnostic testing.
- Self-test on power-up.
- Reset switch.

Software

The Device Server software features include:

- Multiple ways to configure the Device Server:
 - Easy Config Wizard, an easy configuration wizard that allows you to complete basic Device Server configuration
 - DeviceManager, a fully functional Windows 98/NT/2000/ME/Server 2003/XP configuration/management tool
 - WebManager, a web browser option for configuring/managing the Device Server
 - Menu, a window-oriented menu interface for configuration and user access
 - CLI, a Command Line Interface option for configuration/management and user access
 - SNMP, allowing remote configuration via SNMP as well as statistics gathering
 - DHCP/BOOTP, a method of automatically updating the Device Server
- IPv6 support.
- Support for TCP/IP and UDP protocols telnet.
- Virtual modem emulation.
- 'Fixed tty' support for several operating systems (TruePort).
- DHCP/BOOTP for automated network-based setup.
- Dynamic statistics displays and line status reporting for fast problem diagnosis.
- Multi session support on a single terminal.
- Interoperability with IP routing through gateway tables.

Security

The Device Server security features include:

- Supervisory and port (line) password.
- Port locking.
- Per user access level assignment.
- Logging via Syslog.
- Idle port timers, which close a connection that has not been active for a specified period of time.
- Ability to individually disable daemons/services that won't be used by the Device Server.

Supported Products/Versions

Web Browsers

The WebManager has been tested on Windows and Linux with the following web browsers:

- **Netscape**—7.x
- **Internet Explorer**—6.x
- **Mozilla Firefox**—1.x

Typical Applications Summary

Managing the Device Server

The Device Server can be managed and configured by administrators through various methods, allowing them full configuration capabilities and easy access to management statistics and tools. Administrators can access the Device Server using the following methods:

- Connection through ethernet using the DeviceManager, a Windows-based configuration application.
- Connection through ethernet using WebManager, via a web browser.
- Direct connection to the serial port using a Serial Terminal or Terminal Emulation Software.
- From the network through the ethernet interface using reverse Telnet (Port 23).
- Through an SNMP agent, using the Device Server MIB.

Managing/Accessing devices attached to the Device Server

The Device Server can be configured to allow users or administrators to view or manage specific devices on the Device Server's serial port across the Ethernet interface using two different methods.

- **Direct Connect**—users can directly connect to the device on the serial port by Telnet (**Line Service** must be set to **Rev Telnet**) using the Device Server's configured IP address and the serial device's assigned TCP port number.
- **Easy Port Access**—users can connect to the Device Server using the configured Device Server's IP address by reverse Telnet (port number 23), and are provided with a device menu displaying the name of the device that the user has access to. This feature eliminates the need for administrators and users to recall the specific port number associated with a certain device connected to the Device Server. The user can simply connect to a specific device based upon the name of the device and then return to the device menu without disconnecting its initial reverse Telnet connection.

Network Security

The Device Server provides a comprehensive suite of security features to allow an organization to implement robust security planning to prevent unauthorized access. These include trusted host filtering and the ability to disable individual services.



Installation

Introduction

This chapter tells you what is packaged with your IOLAN Device Server, how to power up the Device Server to make sure it works correctly, and how to assign the Device Server an IP address through the LAN.

IOLAN Device Server Components

What's Included

When you open your IOLAN Device Server package, you should have the following components:

- The Device Server
- External power supply (except I/O models)

Note: If the Device Server was bought in bulk, you must supply the power supply.

- *Quick Start Guide* (for I/O models, a soft copy exists on the CD-ROM)
- Warranty Card
- A CD-ROM containing documentation, firmware, DeviceManager, etc.
- Administration cable (consisting of an RJ45-->DB9F adapter and a 3' RJ45 cable) for models that have an RJ45 connector

What You Need to Supply

Before you can begin, you need to have the following:

- A serial cable
- An Ethernet 10/100BASE-T cable if you are connecting the Device Server to the network

Available Accessories

The following accessory is available for purchase for the Device Server:

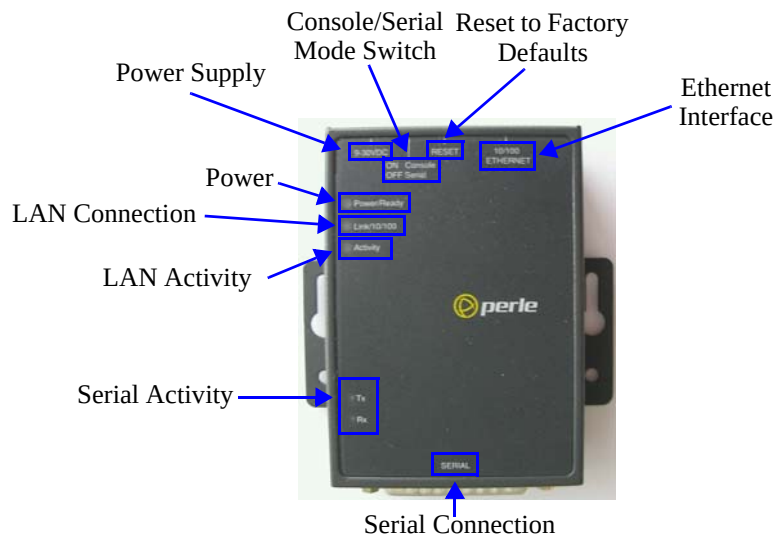
- DIN Rail Mounting Kit (35mm)

Contact your distributor for details.

Getting to Know Your Device Server

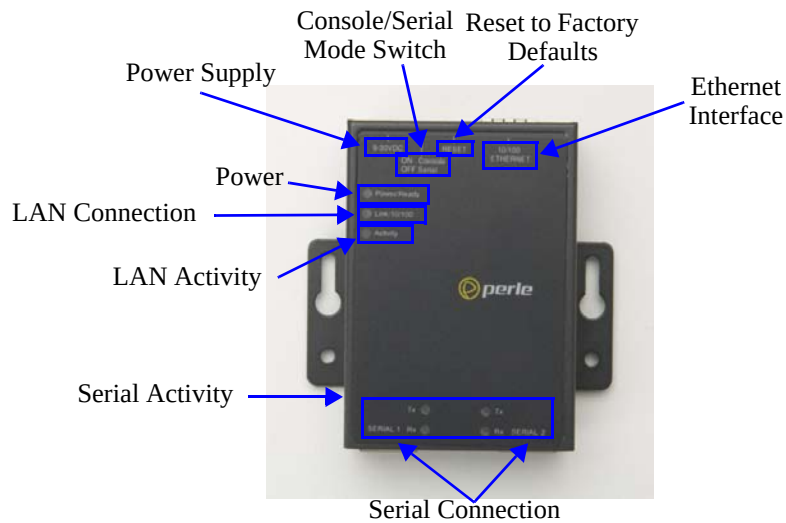
DS1

This section describes the components found on the Device Server.



TS2

This section describes the components found on the TS2 Device Server model.



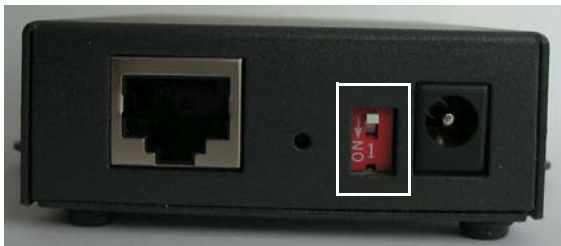
LED Guide

The Device Server LEDs display the following information:

- **Power/Ready**—(Green/Red/Yellow) Shows red at power up. If this LED remains red, indicates that there is a critical error (return to factory). Flashes green to indicate that the Device Server is booting. Flashes green/yellow when the firmware is being updated. Stays solid green to indicate that the Device Server is ready.
- **Link/10/100**
 - Green—10 Mbits
 - Yellow—100 Mbits
 - Off—no LAN connection
- **Activity**—Flashes Green for transmit (TX) or receive (RX) LAN data
- **Tx**—Flashes with transmit serial activity
- **Rx**—Flashes with receive serial activity

Console Mode vs. Serial Mode

You will notice a little switch at the back of the Device Server for switching the Device Server to either Console or Serial mode.



When the switch is down (ON), the Device Server is in Console mode; when the switch is up, the Device Server is in Serial mode. Console mode is used when you have a direct connection between a serial device (like a terminal or a PC) and the Device Server, accessed by the Admin user to configure/manage the Device Server. You can connect directly to the Device Server in Serial mode, but the Device Server will not display all the messages/information you will get in Console mode. Console mode automatically sets the **Serial Interface** to **EIA-232**, **Speed** to **9600**, **Flow Control** to **No**, **Bits** to **8**, **Stop Bits** to **1**, and **Parity** to **None**, in addition to displaying extra system messages. Your Device Server will not work in a production environment in Console mode.

Serial mode is used when the Device Server acts as a communications server, or anytime you are not connecting directly to the Device Server to configure it.

Powering Up the Device Server

Power Supply Requirements

Serial Only Models

If you are providing a power supply for a desktop Device Server model, your power supply must meet the following requirements:

- Output between 9-30V DC.
- The cable attached to the power supply should be about 20AWG, length 6 feet approximately. The barrel dimensions of the cable-plug are OD=5.5, ID=2.1, and length= 9.5mm, with a straight barrel, and positive polarity on the inside and negative polarity on the outside.
- Power can also be provided by pin 12 for DB25M/F and pin 1 RJ45 serial models.

I/O Models

If you are providing a power supply for a desktop Device Server I/O model, your power supply must meet the following requirements:

- Output between 9-30V DC and a minimum of 600mA current.

Note: The maximum load for the Relay channel is 1A @ 30VDC or 0.5A @ 120VAC.

Serial Only Models

Before you attach the Device Server to your network or try to configure it, we suggest that you power it up to verify that it works properly. To power up the Device Server, perform the following steps:

1. Plug the external power supply into the Device Server and then into the electrical outlet.
2. If the Device Server is working correctly, you should see:
 - a. The Power/Ready LED starts out red.
 - b. The Power/Ready LED flashes green while the Device Server boots up.
 - c. The Power/Ready LED stays solid green, indicating that it is ready to configure/use.

You are now ready to begin communicating with your IOLAN Device Server. The last step of the installation process is to set an IP address for the Device Server; this is necessary before it can be configured and put into production.

Before you start to configure the Device Server, you should set the Device Server jumpers if you want to terminate the line.

I/O Models

Before you attach the Device Server to your network or try to configure it, we suggest that you power it up to verify that it works properly. To power up the Device Server, perform the following steps:

1. Unplug the power plugable terminal block from the Device Server.
2. Loosen the screws and then insert your positive (+) wire into the left terminal and screw it down. Insert the negative (-) wire into the right terminal and screw it down.
3. Plug the power terminal block back into the Device Server.
4. Plug the power supply into the electrical outlet.
5. If the Device Server is working correctly, you should see the LEDs cycle for several seconds and then remain a solid green, indicating that it is ready to configure/use.

You are now ready to connect your I/O peripherals to the Device Server and then begin communicating with your IOLAN Device Server. The last step of the installation process is to set an IP address for the Device Server; this is necessary before it can be configured and put into production.

Before you start to configure the Device Server, you should set the Device Server jumpers for Digital I/O (see [Digital I/O Module on page 33](#)) or Analog Input ([Analog Input Module on page 34](#)) channels.

Setting Jumpers

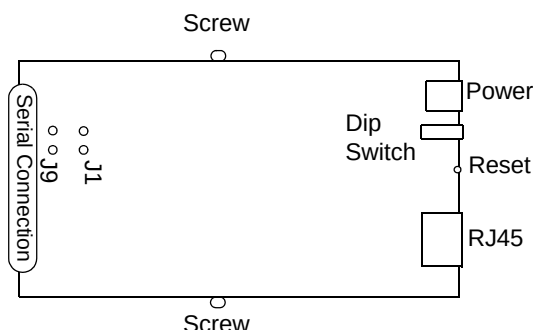
The Device Server contains jumpers that you might need to set before you configure it and put it into production. You can set the power out pin (not on the DB9 model) to a fixed 5V DC output or to the external adapter output; this can range from 9-30V DC (if an external adapter is shipped with the Device Server, it has a 12V DC output). By default, the power out pin is set to no power. You can set the Device Server line termination to **on** or **off** (this is **off** by default) if you are using EIA-422/485 (not applicable for I/O models).

1-Port Device Server

Device Server DB25 Male/Female

To change the settings, do the following:

1. Unplug the Device Server from the electrical outlet and disconnect everything from the box.
2. Open the case by unscrewing the two side screws, one on each side, and lifting off the top of the case. You should see the following:

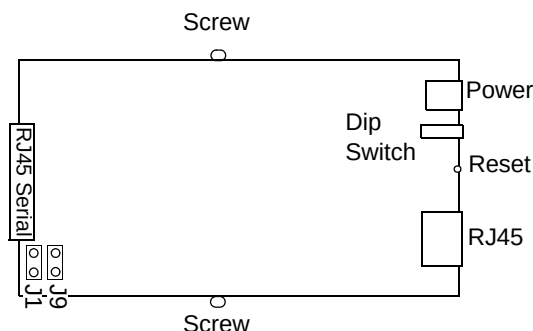


3. To turn line termination **on**, locate and jumper both J1 and J9.
4. Close the Device Server case by replacing the case lid and the two screws. You can now power it on with the new settings.

Device Server RJ45

To change the settings, do the following:

1. Unplug the Device Server from the electrical outlet and disconnect everything from the box.
2. Open the case by unscrewing the two side screws, one on each side, and lifting off the top of the case. You should see the following:

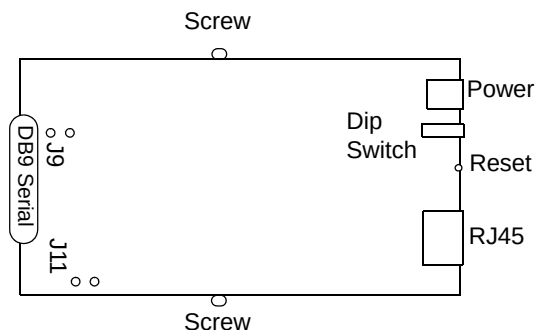


3. To turn line termination **on**, locate and jumper both J1 and J9.
4. Close the Device Server case by replacing the case lid and the two screws. You can now power it on with the new settings.

Device Server DB9

To change the settings, do the following:

1. Unplug the Device Server from the electrical outlet and disconnect everything from the box.
2. Open the case by unscrewing the two side screws, one on each side, and lifting off the top of the case. You should see the following:

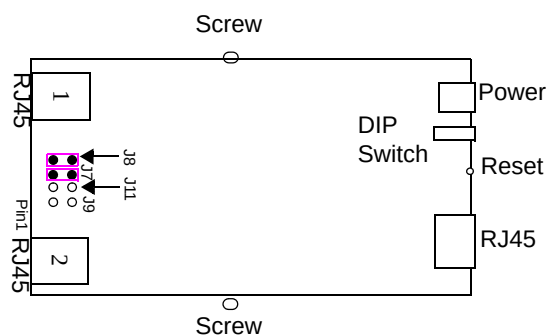


3. To turn line termination **on**, locate and jumper both J11 and J9.
4. Close the Device Server case by replacing the case lid and the two screws. You can now power it on with the new settings.

2-Port Device Server

To change the settings, do the following:

1. Unplug the Device Server from the electrical outlet and disconnect everything from the box.
2. Open the case by unscrewing the two side screws, one on each side, and lifting off the top of the case. You should see the following:

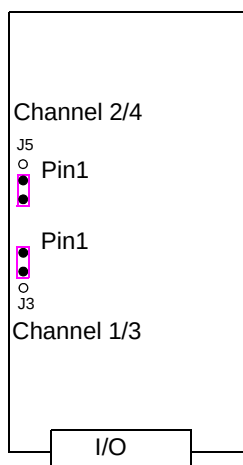


3. To turn line termination **on** for Line 1, locate and jumper both J7 and J8 (as shown in the diagram). To turn line termination **on** for Line 2, locate and jumper both J11 and J9.
4. Close the Device Server case by replacing the case lid and the two screws. You can now power it on with the new settings.

Digital I/O Module

Device Servers that have Digital I/O have an input/output jumper that must be set for each channel and must match the software configuration for each channel. Depending on the model, the placement of the Digital I/O board can change, so the diagram below shows how to set jumper for any Digital board. To change the settings, do the following:

1. Detach the Device Server from the electrical power source and disconnect everything from the box.
2. Open the case by unscrewing the five side screws, two on each side plus the grounding screw, and lifting off the top of the case. You should see the following configuration for the Digital I/O board:



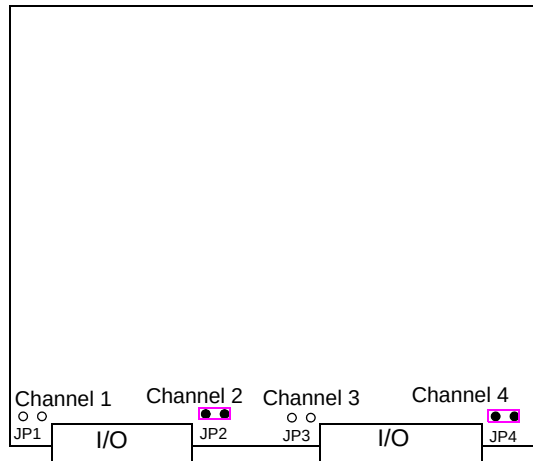
Note: Jumper pins 1 and 2 for Input. Jumper pins 2 and 3 for Output.

3. To configure either Channel 1 or Channel 3 (depending on how many Digital channels your I/O supports and following the mylar channel definitions) for Input, jumper J3 pin 1 and 2 (as shown); this is the default setting. To configure either Channel 2 or Channel 4 (depending on how many Digital channels your I/O supports and following the mylar channel definitions) for Output, jumper J5 pin 2 and 3 (as shown).
4. Close the Device Server case by replacing the case lid and the five screws. You can now power it on with the new settings.

Analog Input Module

Device Servers that have Analog input have a voltage/current jumper that must be set for each channel and must match the software configuration for each channel. To change the settings, do the following:

1. Detach the Device Server from the electrical power source and disconnect everything from the box.
2. Open the case by unscrewing the five side screws, two on each side plus the grounding screw, and lifting off the top of the case. You should see the following configuration for the Analog input board:



3. To configure Channel 1 for Voltage, no jumper should be set (as shown); this is the default setting. To configure Channel 2 for Current, jumper both J2 pins (as shown).
4. Close the Device Server case by replacing the case lid and the five screws. You can now power it on with the new settings.

Wiring I/O Diagrams

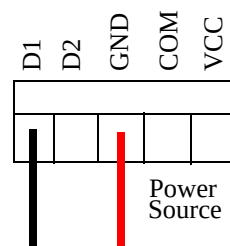
This section describes how to wire the various Device Server I/O models.

Digital I/O

Make sure the Digital I/O jumpers support the software setting; see [Digital I/O Module on page 33](#) for jumper settings.

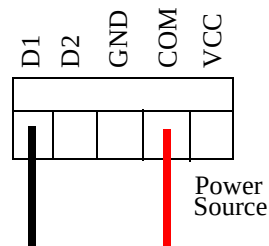
Digital Input Wet Contact

If you are using a wet contact for your Digital input, for channel D1 connect one wire to D1 and the other wire to GND. The power source is supplied by the GND (ground) connector.



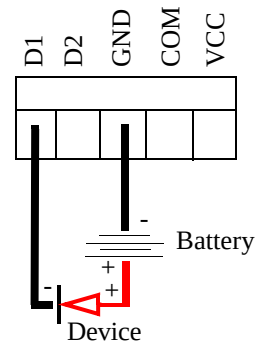
Digital Input Dry Contact

If you are using a dry contact for your Digital input, for channel D1 connect one wire to D1 and the other wire to COM. The power source is supplied by the COM (common) connector.



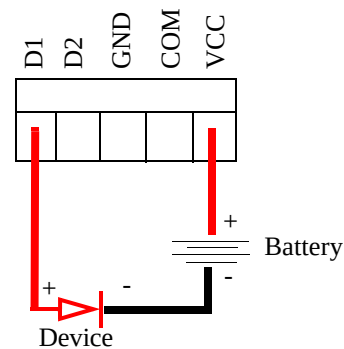
Digital Output Sink

For a Digital output sink (ground) configuration for channel D1, follow the diagram below.



Digital Output Source

For a Digital output source (voltage) configuration for channel D1, follow the diagram below.

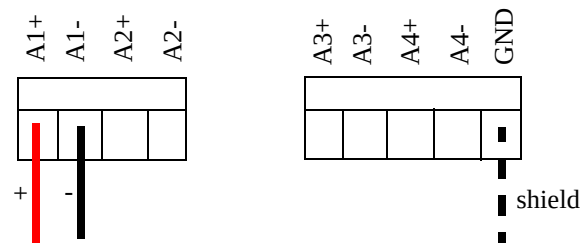


Analog Input

Make sure the Analog jumpers support the software setting; see [Analog Input Module](#) on page 34 for jumper settings.

Current

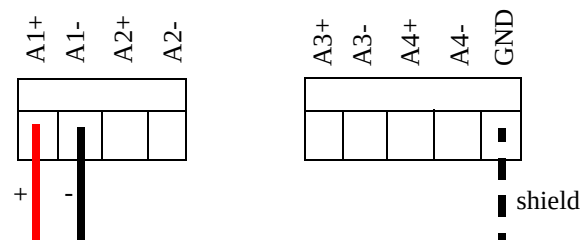
To connect channel A1 with a 2-wire shielded cable, connect the positive wire to A1+, the negative wire to A1-, and optionally the shield to GND.



If you have the positive/negative wires reversed, the output will always read 0 (zero).

Voltage

To connect to Channel A1 with a 2-wire shielded cable, connect the positive wire to A1+, the negative wire to A1-, and optionally the shield to GND.



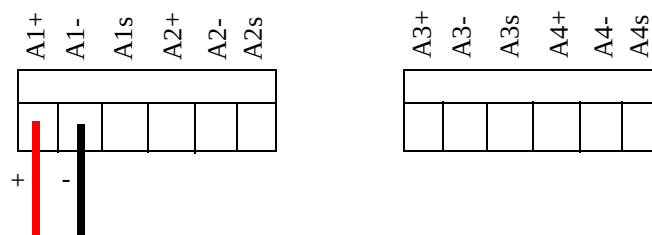
If you have the positive/negative wires reversed, the polarity of the voltage will be reversed.

Temperature Input

If you are using RTD sensors, a short detected status will be displayed if the wires are connected improperly. RTD or thermocouple sensors will display an open detection status when the circuit is broken.

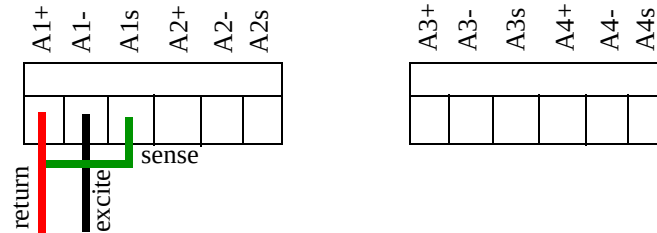
Thermocouple

To connect to Channel A1 with a 2-wire cable, connect the positive wire to A1+ and the negative wire to A1-; you will not be using the A1s connection.



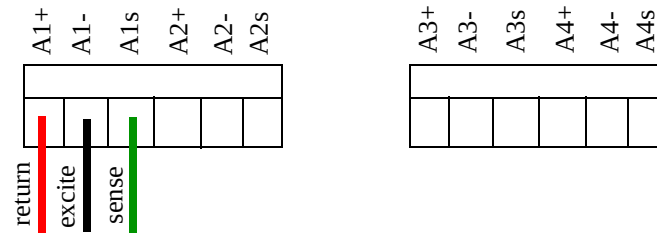
RTD 2-Wire

In a 2-wire RTD configuration, connect the excite wire to A1-, the return wire to A1+, and jumper the sense wire from A1s with a insulated wire going to A1+.



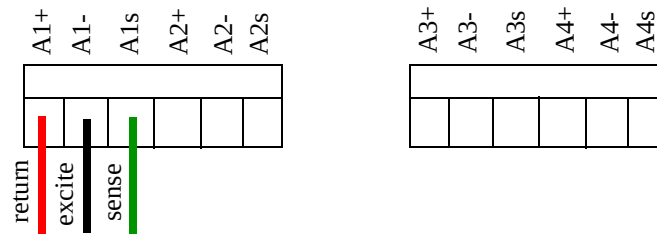
RTD 3-Wire

In a 3-wire RTD configuration, connect the return wire to A1+, the excite wire to A1-, and the sense wire to A1s.



RTD 4-Wire

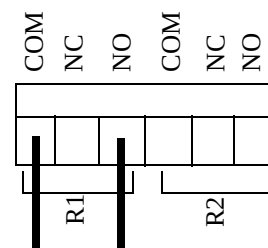
In a 4-wire RTD configuration, connect the return wire to A1+, the excite wire to A1-, the sense wire to A1s, and leave the fourth wire disconnected.



Relay Output

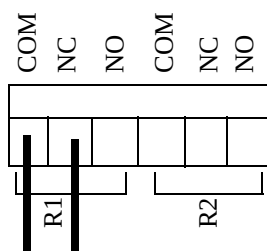
Normally Open Contact

To connect Relay channel R1 for a circuit that is normally inactive, connect one wire to the COM (common) connector and one wire to the NO (normally open) connector.



Normally Closed Contact

To connect relay channel R1 for a circuit that is normally active, connect one wire to the COM (common) connector and one wire to the NC (normally closed) connector.



Setting an Initial IP Address

This section describes the different methods you can use to set the Device Server IP address.

Following is a list of methods for setting the Device Server IP address and a short explanation of when you would want to use that method:

- **Easy Config Wizard**—The Easy Config Wizard is automatically launched from the CD ROM included with your **Device Server**. You can use the Easy Config Wizard to set the Device Server's IP address and configure the line(s).
- **DeviceManager**—Use this method when you can connect the Device Server to the network and access the Device Server from a Windows® PC. The DeviceManager is a Windows-based application that can be used for Device Server configuration and management.
- **Direct Connection**—Use this method when you can connect the Device Server directly to a dumb terminal, essentially logging directly into the Device Server. Using this method, you will need to configure and/or manage the Device Server using either the Menu or CLI.
- **DHCP/BOOTP**—Use this method when you have a BOOTP or DHCP server running and you can connect the Device Server to your network. The Device Server will automatically obtain an IP address from a local network DHCP/BOOTP server when this service is enabled (it is disabled by default).
- **ARP-Ping**—Use this method when you can connect the Device Server to the network and want to assign a temporary IP address to the Device Server by specifying an ARP entry and then pinging it.
- **IPv6 Network**—When the Device Server is connected to an IPv6 network, its local link address should automatically be recognized by the network.

Note: Regardless of which method you use, the Device Server must reside within the same network as the host you are accessing it from.

Once an IP address has been assigned to the Device Server, in most cases, you can continue to use the same method to configure and/or manage the Device Server. See [Chapter 3, Configuration Methods on page 51](#) for more information on the different methods you can use to manage/configure the Device Server.

Using DeviceManager

To use the DeviceManager, you must first install it on a Windows 98/2000/NT/ME/Server 2003/XP operating system (Windows NT requires Service Pack 4 or later) that resides in the same network as the Device Server. The DeviceManager installation wizard can be found on the CD-ROM included in the Device Server package.

1. Connect the Device Server to the LAN and plug it in; it will automatically boot up.
2. From the CD-ROM that was included in the Device Server packaging, select the DeviceManager link.
3. Click on the link under **Location** and click **Open** to automatically start the DeviceManager installation.
4. Install the DeviceManager by following the installation wizard. On the last window, check the **Yes, I want to launch DeviceManager now.** box and click the **Finish** button.
5. On the **Manage Device Server** tab, click the **Search Local Network** button.
6. Any Device Server that does not have an IP address will be displayed as **Not Configured**, with the **Model** and **MAC Address** to identify the Device Server. Highlight the Device Server that you want to assign an IP address to and click the **Assign IP** button.

Note: If your Device Server is displayed with an IP address at this point, you are running a DHCP/BOOTP server on your network and the Device Server has obtained an IP address already. If you want to permanently assign an IP address, continue following the directions, if the DHCP/BOOTP assigned IP address is sufficient, you are now ready to configure the Device Server.

7. Type in the IP address that you want to assign to this Device Server and click the **Assign IP** button.

Note: This is just a temporary IP address that you can use to open a session to the Device Server for configuration.

8. You are now ready to configure the Device Server. Double-click the Device Server you just assigned the temporary IP address to, to open a configuration session. Type **superuser** (the factory default Admin user password) in the Login window and click **OK**.
9. Expand the **Server Configuration** folder and select **Server**. You can choose to enter a permanent IP address in the **Internet Address** field and the **Subnet/Prefix Bits** field of the Server window.

Note: If your network runs a DHCP server and you don't want the Device Server to obtain its **IP Address** from the DHCP server (or if you're not sure if there is a DHCP server and you want to assign a permanent **IP Address**), disable the **DHCP/BOOTP Service** in this window.

10. Click the **Apply** button when you're done with the Server window. To permanently assign the IP address, you need to download the new configuration file and then reboot the Device Server.
11. Download the configuration file to the Device Server by selecting **Tools, Download Configuration to Unit**.
12. Reboot the Device Server by selecting **Tools, Reboot Server**.

For more information on configuring the Device Server using DeviceManager, see [Chapter 5, Using the DeviceManager](#) on page 91.

Using a Direct Connection

You can connect to the Device Server using a PC with a terminal emulation package, such as HyperTerminal or a terminal.

1. Connect the Device Server to your PC or dumb terminal. Make sure the dip switch is in Console mode (this sets the Device Server serial port to EIA-232). When connecting a terminal or PC directly (without modems), the EIA-232 signals need to be crossed over ('null modem' cable). See [EIA-232 Cabling Diagrams on page 46](#) for cabling diagrams.
2. Using a PC emulation application, such as HyperTerminal, or from a dumb terminal, set the Port settings to 9600 Baud, 8 Data bits, No Parity, 1 Stop Bits, and No Hardware Flow control to connect to the Device Server.
3. When prompted, type **admin** for the User and **superuser** for the Password. You should now see the **DS1#** prompt.
4. You are now logged into the Device Server and can set the IP address by typing from the command line using the Command Line Interface (CLI):

```
set server internet <ipv4address>
```

Where **ipv4address** is the IP Address being assigned to the Device Server.
5. Type the following command:

```
save
```
6. If you are going to use another configuration method, such as WebManager or DeviceManager, unplug the Device Server. Change the Device Server dip switch to Off Serial (dip switch in the up position) and connect it to your serial device. Plug the Device Server back in, automatically rebooting the Device Server in the process.
7. If you want to complete the configuration using a direct connection, see [Chapter 3, Configuration Methods on page 51](#) and/or [Chapter 6, Command Line Interface on page 143](#). After you complete configuring the Device Server, unplug the Device Server. Change the Device Server dip switch to Off Serial (dip switch in the up position) and connect it to your serial device. Plug the Device Server back in, automatically rebooting the Device Server in the process.

Using DHCP/BOOTP

If you are using BOOTP, you need to add an entry for the Device Server that associates the MAC address (found on the back of the Device Server) and the IP address that you want to assign to the Device Server. After you have made the MAC address/IP address association for BOOTP, use the following directions for BOOTP or DHCP.

You can connect to the Device Server using a PC with a terminal emulation package, such as HyperTerminal or a terminal.

1. Connect the Device Server to your PC or dumb terminal. Make sure the DIP switch is in Console mode. When connecting a terminal or PC directly (without modems), the EIA-232 signals need to be crossed over ('null modem' cable). See [EIA-232 Cabling Diagrams on page 46](#) for cabling diagrams.
2. Using a PC emulation application, such as HyperTerminal, or from a dumb terminal, set the Port settings to 9600 Baud, 8 Data bits, No Parity, 1 Stop Bits, and No Hardware Flow control to connect to the Device Server.
3. When prompted, type **admin** for the User and **superuser** for the Password. You should now see the a prompt that displays the model type and port number; for example, **DS1#**.
4. You are now logged into the Device Server and can set the IP address by typing from the command line using the Command Line Interface (CLI). Type the following command:

```
set server service dhcp/bootp on
```

5. Type the following command:
`save`
6. Type the following command:
`reboot`
7. When the Device Server reboots, it will automatically poll for an IP address from the DHCP/BOOTP server. If you have a Device Server with dual Ethernet, each Ethernet connection will automatically be assigned an IP address, you can access the Device Server through either IP address.

If for some reason it cannot obtain an IP address from your DHCP/BOOTP server, you will have to either connect to the Device Server on the console port and reboot it or push the Reset to Factory button to access the Device Server.

You are now ready to configure the Device Server. See [Chapter 3, Configuration Methods](#) on page 51 for information on the different Device Server configuration methods.

Using ARP-Ping

You can use the ARP-Ping (Address Resolution Protocol) method to temporarily assign an IP address and connect to your Device Server to assign a permanent IP address. To use ARP-Ping to temporarily assign an IP address:

1. From a local UNIX/Linux host, type the following:
`arp -s a.b.c.d aa:bb:cc:dd:ee:ff`

On a Windows® 98 or newer system, type the following:

`arp -s a.b.c.d aa-bb-cc-dd-ee-ff`

(where **a.b.c.d** is the IPv4 address you want to temporarily assign to the Device Server, and **aa:bb:cc:dd:ee:ff** is the Ethernet (MAC) address of Device Server, found on the back of the unit.

2. Whether you use UNIX or Windows®, you are now ready to ping to the Device Server. Here is a UNIX example of the sequence to use:
`arp -s 192.168.209.8 00:80:d4:00:33:4e`
`ping 192.168.209.8`

You are now ready to configure the Device Server. See [Chapter 3, Configuration Methods](#) on page 51 for information on the different Device Server configuration methods.

IPv6 Network

The Device Server has a factory default link local IPv6 address that takes the following format:

Device Server MAC Address: 00-80-D4-AB-CD-EF

Link Local Address: fe80::0280:D4ff:feAB:CDEF

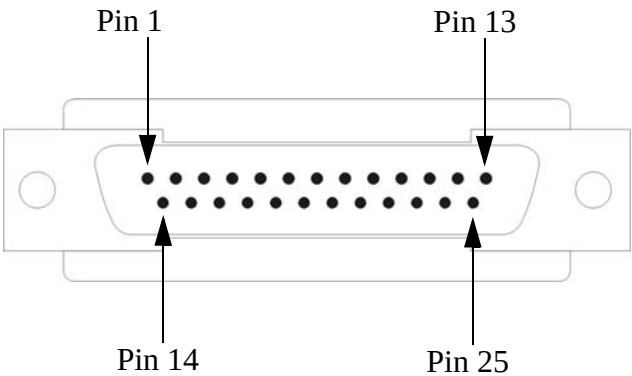
The Device Server will also listen for IPv6 router advertisements to learn a global address. You do not need to configure an IPv4 address for a Device Server residing in an IPv6 network.

You are now ready to configure the Device Server. See [Chapter 3, Configuration Methods](#) on page 51 for information on the different Device Server configuration methods.

Pinouts

DB25 Male

This section defines the pinouts for the DB25 male connection used on the 1-port Device Server.



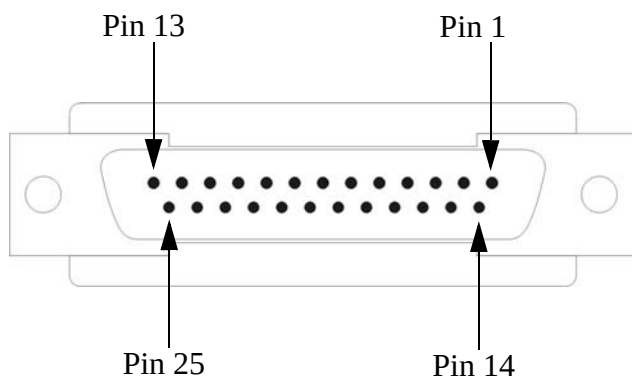
The following table provides pinout information:

Pinout	EIA-232	EIA-422	EIA-485 Full Duplex	EIA-485 Half Duplex
1	Shield	Shield	Shield	Shield
2 (out)	TxD			
3 (in)	RxD			
4 (out)	RTS			
5 (in)	CTS			
6 (in)	DSR			
7	GND	GND	GND	GND
8 (in)	DCD			
12	Power in	Power in	Power in	Power in
13		CTS-		
14		TxD+	TxD+	DATA+
15		TxD-	TxD-	DATA-
18		RTS+		
19		RTS-		
20 (out)	DTR			
21		RxD+	RxD+	
22		RxD-	RxD-	
25		CTS+		

The power in pin, pin 12, can be 9-30V DC.

DB25 Female

This section defines the pinouts for the DB25 female connection used on the 1-port Device Server.



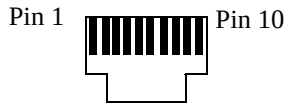
The following table provides pinout information:

Pinout	EIA-232	EIA-422	EIA-485 Full Duplex	EIA-485 Half Duplex
1	Shield	Shield	Shield	Shield
2 (out)	RxD			
3 (in)	TxD			
4 (out)	CTS			
5 (in)	RTS			
6 (in)	DTR			
7	GND	GND	GND	GND
8 (in)	DCD			
12	Power in	Power in	Power in	Power in
13		RTS-		
14		RxD+	RxD+	DATA+
15		RxD-	RxD-	DATA-
18		CTS+		
19		CTS-		
20 (out)	DSR			
21		TxD+	TxD+	
22		TxD-	TxD-	
25		RTS+		

The power in pin, pin 12, can be 9-30V DC.

RJ45

This section defines the pinouts for the RJ45 connection used on the DS and TS Device Server. The TS Device Server does not support power in, so use the 8-pin mappings for this model. These pinouts do not apply to I/O models.



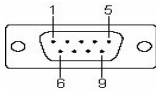
The following table provides pinout information:

Pinout 10-pin	Pinout 8-pin	EIA-232	EIA-422	EIA-485 Full Duplex	EIA-485 Half Duplex
1		Power In	Power In	Power In	Power In
2 (in)	1	DCD			
3 (out)	2	RTS	TxD+	TxD	TxD+/RxD+
4 (in)	3	DSR			
5 (out)	4	TxD	TxD-	TxD-	TxD-/RxD-
6 (in)	5	RxD	RxD+	RxD+	
7	6	GND	GND	GND	GND
8 (in)	7	CTS	RxD-	RxD-	

The power in pin, Pin 1, can be 9-30V DC.

DB9 Male (Serial Only)

This section defines the pinouts for the DB9 male connection used on the 1-port Device Server that is serial only (not I/O).

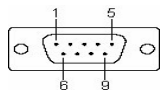


The following table provides pinout information:

Pinout 9-pin	EIA-232	EIA-422/485 Full Duplex	EIA-485 Half Duplex
1 (in)	DCD		
2 (in)	RxD	RxD+	RxD+
3 (out)	TxD	TxD+	TxD+
4 (out)	DTR		
5	GND	GND	GND
6 (in)	DSR	RxD-	RxD-
7	RTS		
8 (in)	CTS		
9		TxD-	TxD-

DB9 Male I/O

This section defines the pinouts for the DB9 male connection used on the 1-port Device Server I/O models.



The following table provides pinout information:

Pinout 9-pin	EIA-232	EIA-422/485 Full Duplex	EIA-485 Half Duplex
1(in)	DCD		
2 (in)	RxD	RxD+	
3 (out)	TxD	TxD-	TxD-/RxD-
4 (out)	DTR		
5	GND	GND	GND
6 (in)	DSR	RxD-	
7	RTS	TxD+	TxD+/RxD+
8 (in)	CTS		
9			

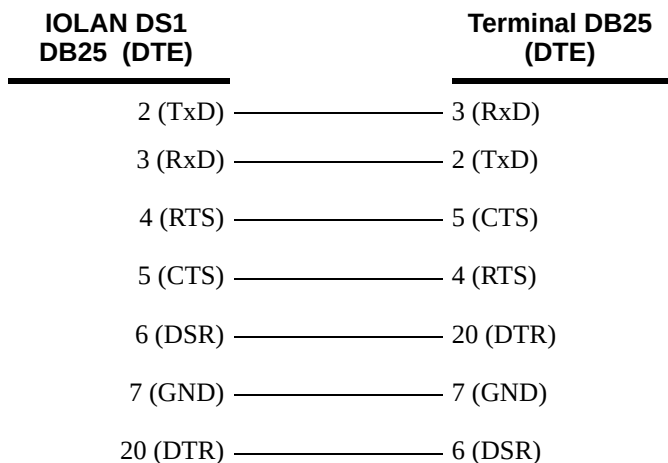
EIA-232 Cabling Diagrams

This section shows how to create EIA-232 cables that are compatible with the Device Server.

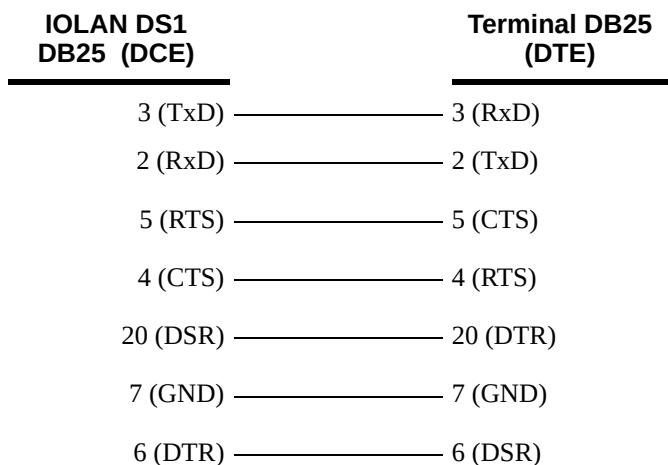
Terminal DB25 Connector

The following diagrams show how the null modem cable should be configured when connecting to a terminal DB25.

DB25 Male



DB25 Female



RJ45

IOLAN RJ45		Terminal DB25 (DTE)	
10-pin	8-pin		
4 (DSR)	3	—————	20 (DTR)
3 (RTS)	2	—————	5 (CTS)
5 (TxD)	4	—————	3 (RxD)
6 (RxD)	5	—————	2 (TxD)
7 (GND)	6	—————	7 (GND)
8 (CTS)	7	—————	4 (RTS)
9 (DTR)	8	—————	6 (DSR)

DB9 Male

IOLAN DS1 DB9 Male	Terminal DB25 (DTE)
3 (TxD)	————— 3 (RxD)
2 (RxD)	————— 2 (TxD)
7 (RTS)	————— 5 (CTS)
8 (CTS)	————— 4 (RTS)
6 (DSR)	————— 20 (DTR)
5 (GND)	————— 7 (GND)
4 (DTR)	————— 6 (DSR)

Modem DB25 Connector

The following diagrams show how a standard straight through cable should be configured when connecting to a DB25 modem.

DB25 Male

IOLAN DS1 DB25 (DTE)		Modem DB25 (DCE)
2 (TxD)	—————	2 (RxD)
3 (RxD)	—————	3 (TxD)
4 (RTS)	—————	4 (CTS)
5 (CTS)	—————	5 (RTS)
6 (DSR)	—————	6 (DSR)
7 (GND)	—————	7 (GND)
8 (DCD)	—————	8 (DCD)
20 (DTR)	—————	20 (DTR)

RJ45

IOLAN RJ45			Modem DB25 (DCE)
10-pin	8-pin		
2 (DCD)	1	—————	8 (DCD)
3 (RTS)	2	—————	4 (CTS)
4 (DSR)	3	—————	6 (DSR)
5 (TxD)	4	—————	2 (RxD)
6 (RxD)	5	—————	3 (TxD)
7 (GND)	6	—————	7 (GND)
8 (CTS)	7	—————	5 (RTS)
9 (DTR)	8	—————	20 (DTR)

DB9 Male

IOLAN DS1 DB9 Male	Modem DB25 (DCE)
1 (DCD)	8 (DCD)
2 (RxD)	3 (TxD)
3 (TxD)	2 (RxD)
4 (DTR)	20 (DTR)
5 (GND)	7 (GND)
6 (DSR)	6 (DSR)
7 (RTS)	4 (CTS)
8 (CTS)	5 (RTS)



Configuration Methods

Introduction

This chapter provides information about the different methods you can use to configure the Device Server. Before you can configure the Device Server, you must assign an IP address to the Device Server. You can assign an IP address to the Device Server using one of the following methods:

- Using the DeviceManager as described in [Using DeviceManager on page 39](#).
- Using ARP-Ping as described in [Using ARP-Ping on page 41](#).
- Using a direct connection to the Admin port as described in [Using a Direct Connection on page 40](#).

DeviceManager

The DeviceManager is a fully functional Windows 98/NT/2000/ME/Server 2003/XP Device Server configuration/management tool. You must install the DeviceManager from the CD-ROM included with the Device Server. Through the DeviceManager, you can:

- assign an IP address to new Device Servers.
- perform firmware updates.
- create configuration files, which can be immediately downloaded to the Device Server.
- save configuration files locally in the DeviceManager's native binary format or to a text file. The text configuration file can be edited with a text editor.
- open a session to a Device Server and import a (saved) configuration file.
- view statistics for a Device Server.
- download custom files, such as new terminal definitions and a custom language file.
- download a configuration file to multiple Device Servers.

You can use the DeviceManager as a stand-alone application to create configuration files that can be saved locally or you can use the DeviceManager to open a session to a Device Server to actively manage and configure it.

See [Chapter 5, Using the DeviceManager on page 91](#) for information on configuring/managing the Device Server with DeviceManager.

WebManager

The WebManager is a web-browser based method of configuring/managing a Device Server.

To access a Device Server through the WebManager, open up your web browser and type in the IP address of the Device Server that you want to manage/configure. A login screen will appear.type in the Admin password.

Using the WebManager

The Server Configuration window is displayed after you first log on. The running Device Server configuration is displayed in the WebManager. You navigate through the different configuration windows by selecting the configuration window from the drop-down options in the upper-lefthand corner of the browser.

When you have completed all the changes to a configuration window, click the **Submit** button. After you make all your configuration changes, click the **Save to FLASH** button. If you want your changes to take effect immediately, click the **Reboot** button. You can make changes to a line, **Submit** them, and then click the **Kill Line** button to test the changes immediately; however, if you do not click the **Save to FLASH** button, your changes will be lost the next time the Device Server reboots. After you click the **Reboot** button, you will need to reconnect and login to the Device Server.

Note: Use the WebManager's drop-down menus to navigate through the WebManager. Do not use the browser's Back button.

CLI

The Command Line Interface (CLI) is a command line option for Device Server configuration/management and user access. See [Chapter 6, *Command Line Interface* on page 143](#) for a full explanation of how to use the CLI.

Menu

The Menu is a window-oriented Device Server configuration and user access option. To manage the Device Server, you will also need to use the CLI, WebManager, or DeviceManager, as you cannot download or upload files to the Device Server through the Menu.

Accessing the Menu

Menu access is available to any user whose Line Service is set to DSLogin, and whose User Service is set to DSPrompt. What the user sees depends on what the User Level is set to:

- **Menu**—Users with **User Level Menu** will only see the sessions that have been set up for them. They can start predefined sessions, kill (stop) a running session, resume a session, and logout of the Device Server.
- **Restricted**—Users with **User Level Restricted** can basically perform the same tasks as a Menu user, except that they have the option of performing these tasks via the Menu or the CLI.
- **Normal**—Users with **User Level Normal** can do everything a Restricted user can do, plus start a free session (connecting to any host on the network), set up their own user parameters (sessions, password, language, hotkey prefix), define their terminal, and become the Admin user (if they know the Admin password).
- **Admin**—Users with **User Level Admin** (not the Admin user), have complete access to the Device Server, the same as the Admin user. Through the Menu program, the Admin level user can configure the Device Server, although there are several tasks that can only be done in the CLI, such as downloading and uploading files and saving the configuration to FLASH.

Menu Conventions

You select an option from the Menu by using the keyboard up and down arrows to navigate the list. When the menu item you want to access is highlighted, press the **Enter** key to either get to the next list of options or to get the configuration screen, depending on what you select. When you are done configuring parameters in a screen, press the **Enter** key and then the **Enter** key again to **Accept and exit the form**. If you want to discard your changes, press the **Esc** key to exit a screen, at which point you will be prompted with **Changes will be lost, proceed? (y/n)**, type **y** to discard your changes or **n** to return to the screen so you can press **Enter** to submit your changes.

If there are a number of predefined options available for a field, you can scroll through those items by pressing the **Space Bar** or you can type **l** (lowercase L) to get a list of options, use the up/down arrows to highlight the option you want, and then press **Enter** to select it.

DHCP/BOOTP

If you have a DHCP/BOOTP server and the Device Server's Server Service DHCP/BOOTP is enabled, the Device Server can obtain its IP address and several configuration parameters from the DHCP/BOOTP server when it boots up. However, you must use another method for creating the configuration file, like the DeviceManager, WebManager, or the CLI. See [DHCP/BOOTP Parameters on page 74](#) for more information on the DHCP/BOOTP parameters that can be set for the Device Server.

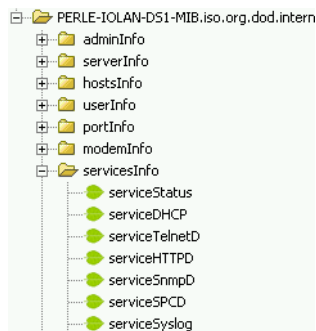
When DHCP/BOOTP is enabled and there is a DHCP/BOOTP server within the network, the IP Address obtained from DHCP/BOOTP will always override the Device Server's configured IP Address when the Device Server is rebooted.

SNMP

Before you can configure/manage the Device Server using SNMP, you need to set the Device Server IP address and configure a read-write user for SNMP version 3 or a community for SNMP version 1 or 2. You can use DeviceManager, CLI, or the Menu to set the IP address and user/community (don't forget to reboot the Device Server before connecting with the SNMP manager to make your changes take effect).

Once the IP address and user/community have been set, load the file from the Device Server CD-ROM into your SNMP manager.

Connect to the Device Server through your SNMP manager using its IP address to configure/manage the Device Server. Expand the **PERLE-IOLAN-DS1-MIB** folder to see the Device Server's parameter folders. Below is an example of the configurable parameters under the **ServicesInfo** folder.



The first variable in each folder is the **Status** variable, for example, **serviceStatus**. When you perform a **GET** on this variable, one of the following values will be returned:

- **1**—Indicates that the container folder is active with no changes.
- **2**—Indicates that the container folder is active with change(s).

Once you have completed setting the variables in a folder, you will want to submit your changes to the Device Server. To do this, set the **Status** variable to **4**. If you want to discard the changes, set the **Status** variable to **6**.

- **4**—Indicates that the changes in the container folder are to be submitted to the Device Server.
- **6**—Indicates that the changes in the container folder are to be discarded.

If you want to save all the changes that have been submitted to the Device Server, you need to expand the **adminInfo** container folder and **SET** the **adminFunction** to **1** to write to FLASH. To make the configuration changes take effect, **SET** the **adminFunction** to **3** to reboot the Device Server.



Configuring the Device Server

Introduction

This chapter provides general information about configuring the Device Server for your production environment. Although this chapter is not specific to any configuration method, there should be enough information that you can apply the information to any of the configuration methods.

When you are configuring the Device Server, remember that none of your configuration changes will be permanent until you submit/apply your changes, save to FLASH, and reboot the Device Server.

Configuring the Device Server

General Device Server Configuration

At this point, you should already have assigned the Device Server an IP address. Therefore, you have your choice of how to configure the Device Server; using the DeviceManager, WebManager, Menu, CLI, or SNMP.

Device Server Services

In order to be as flexible and accessible as the Device Server is, it can run several predefined daemon and client applications. The Device Server can run the following daemon applications:

- TelnetD
- SPCD (the TruePort daemon)
- DeviceManagerD
- HTTPD
- SNMPD
- ModbusD

If you disable any of the daemons, it can affect how the Device Server can be used or accessed. For example, if you disable HTTPD, you will not be able to access the Device Server with the WebManager. If you disable DeviceManagerD, the DeviceManager will not be able to connect to the Device Server. If you do not want to allow users to Telnet to the Device Server, you can disable TelnetD; therefore, disabling daemons can also be used as an added security method for accessing the Device Server.

The following client applications can run on the Device Server:

- Syslog
- DHCP/BOOTP

If you do not have a DHCP/BOOTP server in your network, we recommend that you disable the DHCP/BOOTP service to speed up Device Server reboots (otherwise, the Device Server waits for a DHCP/BOOTP packet until it times out on a reboot).

By default, all daemon and client applications (except DHCP/BOOTP) are enabled and running on the Device Server.

TruePort

The TruePort utility acts as a com port redirector that allows applications to talk to serial devices across a network as though the serial devices were directly attached to the server. For Device Server I/O models, you can also monitor and control I/O through the TruePort client. You can map the baud rate of the host COM port to a higher baud rate for the serial line that connects the serial device and the Device Server. You must be running the TruePort daemon on the host that is accessing the serial device for this to work. See [TruePort on page 205](#) for more information about the TruePort utility.

Hardware Configuration

Configure the ethernet interface that is connecting the Device Server to the LAN and the serial cable that is connecting the Device Server to the serial device.

Ethernet Connection

You need to know the ethernet interface speed and duplex as follows, unless you are using the Auto detect option:

- 10 Mbps half or full duplex
- 100 Mbps half or full duplex

Serial Connection

You also need to know the serial interface specifications as follows (TS2 support EIA-232 only):

- EIA-232 and its speed
- EIA-422 and its speed
- EIA-485 and
 - its speed
 - half duplex with/without echo suppression or full duplex
 - TX driver control is automatic or RTS

Other

The most important thing to keep in mind when configuring the hardware parameters is to make sure that they are consistent with the serial device you have connected to the port. So, if you are connecting to a modem that sends out a DSR signal, you probably want to turn the **Monitor DSR** option on. Following is a list of just some of the other hardware configuration options:

- Data Bits—5 to 8
- Stop Bits—1, 1.5, 2
- Monitor DSR—on, off
- Monitor DCD—on, off
- Parity—None, Odd, Even, Space, Mark
- Flow—Software, Hardware, or None

Modbus Configuration

This section provides a brief overview of the steps required to configure a Device Server for your Modbus environment. You can read the [Modbus Gateway Settings on page 58](#) and [Modbus Line Settings on page 59](#) sections for more specific information about the Modbus settings.

Overview

Configuring a Master Gateway

To configure a Master Gateway (Modbus Master resides on the serial side of the Device Server), do the following:

1. Verify that the default Modbus Gateway settings (the settings to the Slave Gateway do not apply here) in the Server section work in your environment; if they don't configure as required.
2. Set the **Line Service** parameter to **Modbus Master** for the Line connected to the Modbus serial Master.
3. In the Modbus Master settings, map the Modbus TCP Slave's IP addresses and their UIDs that the Modbus serial Master will attempt to communicate with.

Configuring a Slave Gateway

To configure a Slave Gateway (Modbus Master resides on the TCP/Ethernet network), do the following:

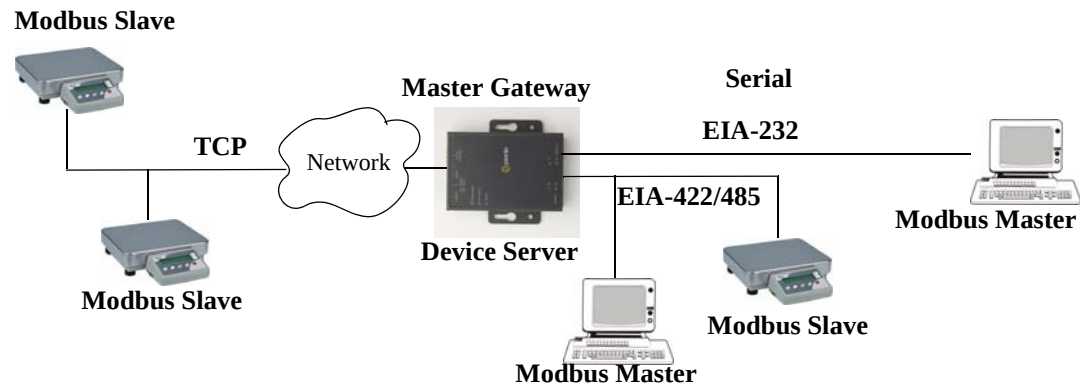
1. Verify that all the default Modbus Gateway settings in the Server section work in your environment; if they don't configure as required.
2. Set the **Line Service** parameter to **Modbus Slave** for the Line connected to the Modbus serial Slaves.
3. In the Modbus Slave settings, specify the Modbus Slave UIDs that the Modbus TCP Master will attempt to communicate with.

Modbus Gateway Settings

The scenarios in this section are used to illustrate how the Modbus Gateway settings are incorporated into a Modbus device environment. Depending on how your Modbus Master or Slave devices are distributed, the Device Server can act as both a Slave and Master Gateway(s) on a multiport Device Server or as either a Slave or Master Gateway on a single port Device Server.

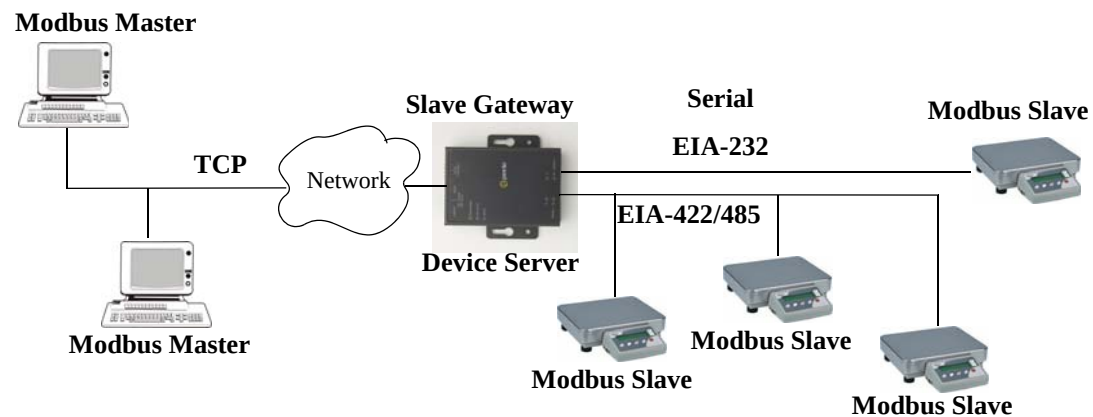
Modbus Master Gateway

The Device Server acts as a Master Gateway when the Modbus Master resides on the serial side of the Device Server. Each Modbus Master can communicate to UIDs 1-247.



Modbus Slave Gateway

The Device Server acts as a Slave Gateway when the Modbus Master resides on the TCP/Ethernet network and the Modbus Slaves reside on the serial side of the Device Server. Note that there is only one Slave Gateway for the Device Server. You can define only one Slave Gateway for the Device Server, although multiple lines/ports can participate as part of that gateway (depending on how you configure the **Line Service** settings).

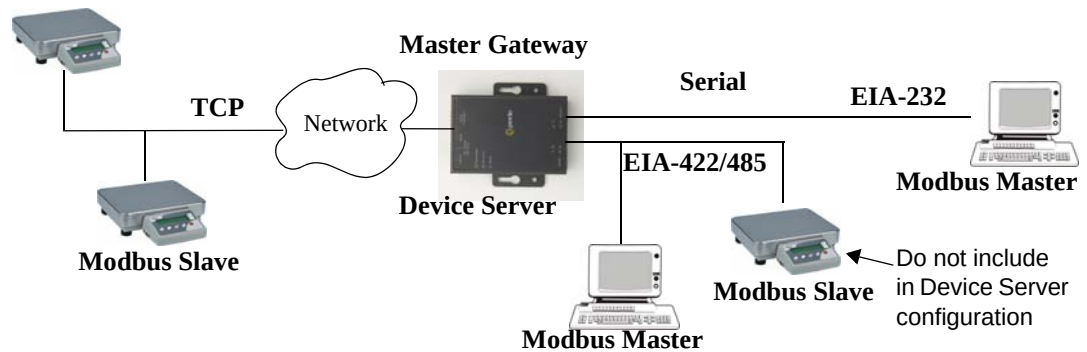


Modbus Line Settings

Modbus Master Settings

When you have Modbus Masters on the serial side of the Device Server, configure the Line as a Modbus Master. If you also have a Modbus serial Slave on the same serial network as the serial Modbus Master that communicates with that serial Master, do not define its UID in the Remote Slave IP Mappings settings, or the Modbus serial Slave may not function properly. You must configure the Modbus TCP Slaves (we term these as Remote Slave IP Mappings) on the TCP/Ethernet side so the Device Server can properly route messages to the appropriate UIDs configured for those remote Modbus TCP Slaves.

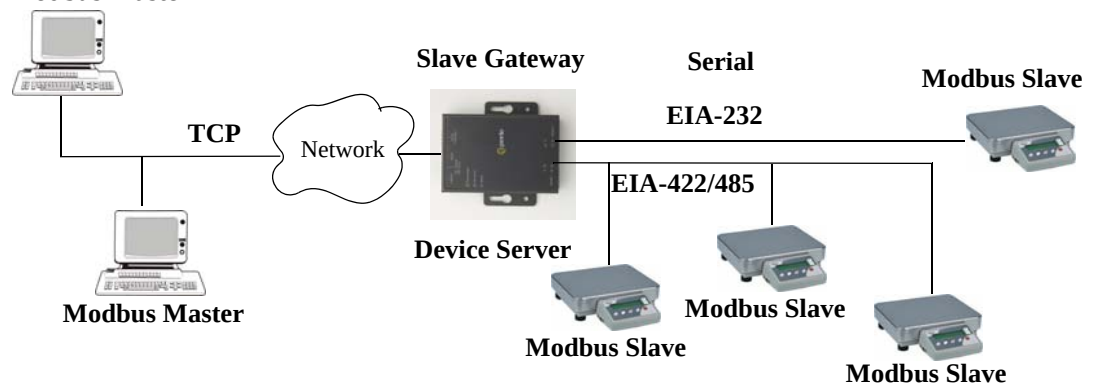
Modbus Slave



Modbus Slave Settings

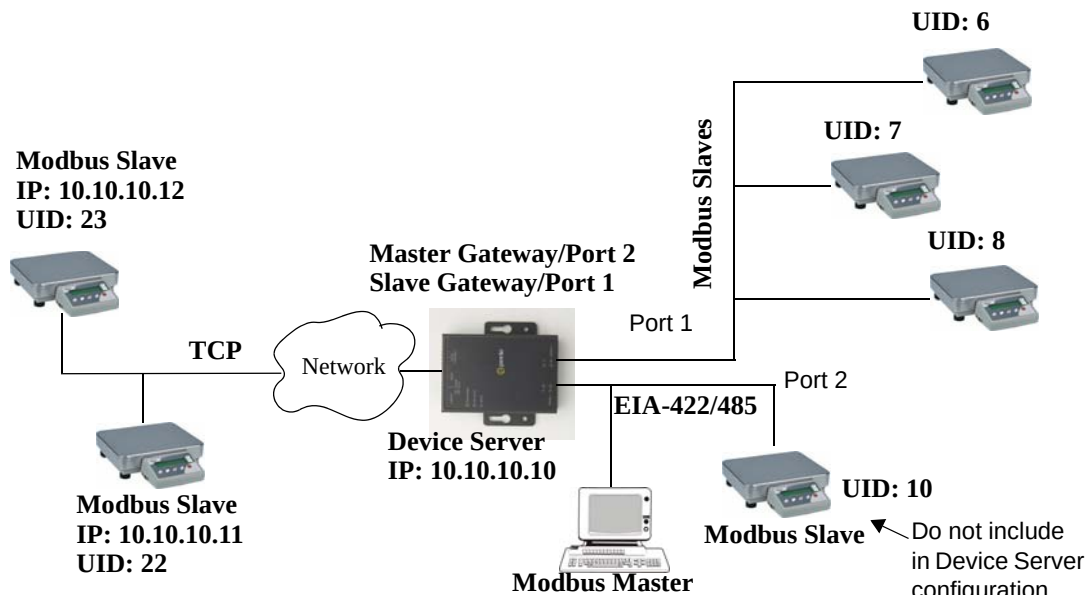
When you have Modbus Slaves on the serial side of the Device Server, configure the Line as a Modbus Slave. There is only one Slave Gateway in the Device Server, so all Modbus serial Slaves must be configured uniquely for that one Slave Gateway; all Modbus serial Slaves must have unique UIDs, even if they reside on different serial ports, because they all must be configured to communicate through the one Slave Gateway.

Modbus Master

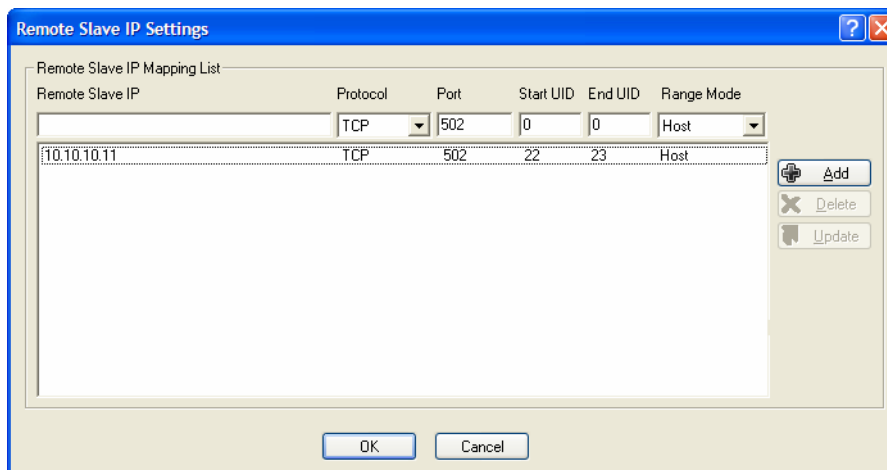


Example Scenario

The following example describes the settings that you would configure to set up a Modbus environment on a multiport Device Server, where the Modbus Master resides on a serial port/line connected to the Device Server. This scenario assumes two things, that the **Service ModbusD** (the Modbus daemon) is enabled and that the default **Modbus Gateway** settings have not been changed. The Modbus Master communicates with Modbus Slaves that reside on the TCP/Ethernet network and on another serial port defined as part of the Slave Gateway in the Device Server, and with a Modbus serial Slave (UID 10) that is on the same serial line as the Modbus Master itself. The Device Server will act as a Master Gateway for the Modbus serial Master and allow it to communicate to the remote Modbus TCP Slaves. By configuring the Device Server's own IP address as a remote Modbus Slave and having the Slave Gateway configured, the Modbus serial Master can communicate with Modbus serial Slaves on another serial port/line on the Device Server (UIDs 6-8).



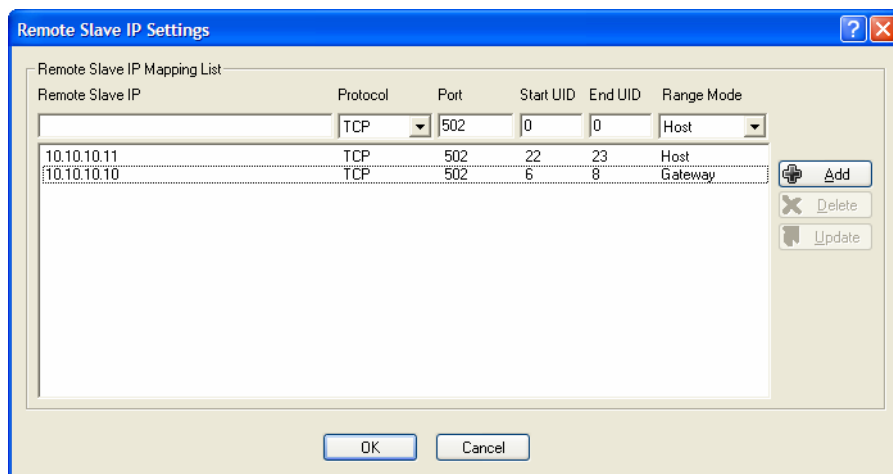
When the Modbus Master is communicating with the TCP/Ethernet Modbus Slaves, the Line/port that the Modbus Master is attached to must be configured with a **Line Service** of **Modbus Master**. By configuring the Remote Slave IP settings as:



The Device Server will send a request and expect a response from a Modbus Slave with an IP Address of 10.10.10.11 on Port 502 with UID 22 and from Modbus Slave with IP Address of 10.10.10.12 on Port 502 with UID 23 (remember when **Range Mode** is set to **Host**, the Device Server increments the last octet of the IP address for each UID specified in the range).

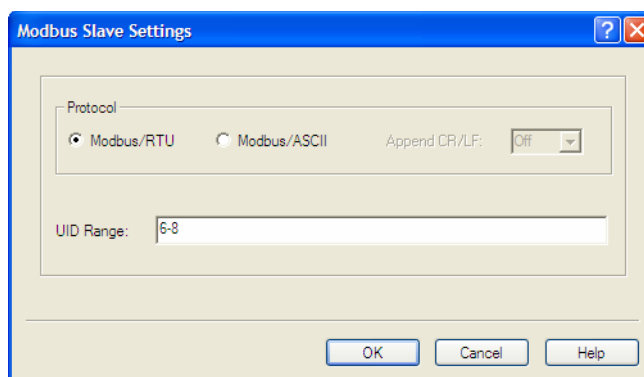
Also note that the Modbus Slave (UID 10) that is on the same line as the Modbus Master should not be configured anywhere in the Device Server's Master Gateway table for that serial port, or a Modbus Exception may be sent (if this option is enabled) because the Device Server will attempt to connect and send to a non-existent remote Modbus TCP Slave and a response timeout can occur.

To communicate with the Modbus Slaves on the serial side of the Device Server, the Device Server must also be configured to be a Slave Gateway. The Modbus Slaves on a serial port attached to the Device Server must be connected to a Line/port that is configured for the **Line Service** of **Modbus Slave**. To communicate with the Modbus Slaves on the serial port configured as part of a Slave Gateway, the Remote Slave IP settings are configured as:



The Device Server also acts as a Slave Modbus Gateway, receiving all the messages for IP address 10.10.10.10 and routing them to Modbus Slave devices with UIDs 6, 7, and 8.

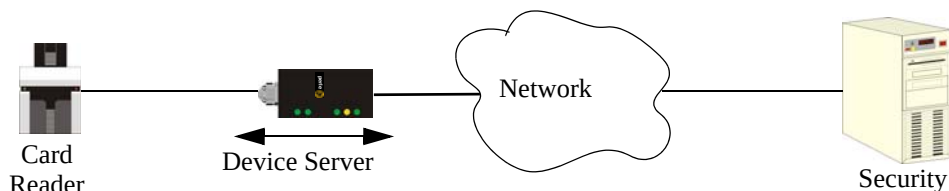
You must also configure the **Line Service** as **Modbus Slave** for the Modbus serial slaves as:



The Modbus serial Master will attempt to communicate through the Modbus Master Gateway to Modbus serial Slaves with UIDs 6, 7, and 8. In order to accomplish this, the communication is routed through the Device Server's Modbus Slave Gateway from the Device Server's Modbus Master Gateway, to the serial Slaves.

Machine To Machine Connections

If you are using the Device Server to connect two hosts, allowing data to flow freely between them, you just need to configure the **Server** and the **Line** (no **User** required). In the following example, the serial device is a security Card Reader that needs to transmit and receive information to/from a host on the network that maintains the Card Reader's application every time an employee uses an access card to attempt to gain entry to the company.



After configuring the **Server** parameters (**Server Name**, **IP Address**, **Ethernet** and **Serial** interfaces, etc.), the **Line Service** is set to **Sil Raw**, which creates an automatic, continuous connection between the Card Reader and its associated application on the Security host (though the Device Server), by specifying the Security host name (which must already be configured in the Device Server's Host Table) and TCP/IP port number. Therefore, the Card Reader can make a request to the Security host card reader application for employee verification, also logging access time, employee name, etc., and the Security host application can send back a code that does or does not unlock the door.

Users Connecting to Serial Devices

For a user to connect to the serial device connected to the Device Server from the LAN, the **Line Service** must be set to **Rev Telnet**. The user will either access the serial device directly or go through the Easy Port Access Menu, depending on the **User Level** setting.

Users who are **Level Admin** or **Normal** will access the serial device directly; the user must connect to the Device Server's IP address and port number (the **DS Port** parameter). The user will be asked to login with a user name and password; if this is successful, the user is automatically connected to the serial device.

Users who are **Level Restricted** or **Menu** can access the serial device through the Easy Port Access Menu, which displays the line number and name and a logout option; the user just needs to connect to the Device Server's IP address. The user will be asked to login with a user name and password; if this is successful, the Easy Port Access Menu is displayed.

Users Connecting to the LAN

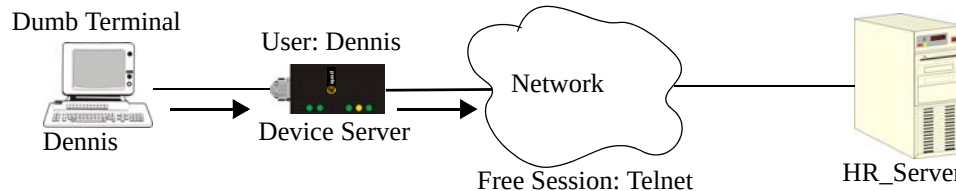
For a user to connect to the LAN through the Device Server from a serial device, the **Line Service** can be set to any **Direct** or **Silent** setting, plus **Bidir** or **DSLogin**.

User accounts should be created when:

- authentication is being done locally by the Device Server.
- you want to create predefined sessions for a user to limit that user's access to the network.

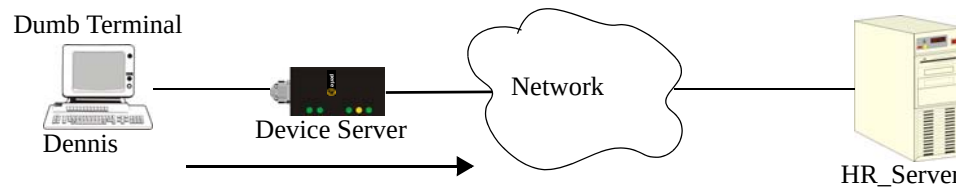
Connecting To the Device Server

When a user connects to the Device Server, that user is authenticated and is usually set up with predefined sessions or given the opportunity to configure a **Free Session** to access any host using any protocol (must have a **Level** of at least **Normal** to configure a **Free Session**). In this example, the user must have a **Line** and **User Service** of **DSLogin** and **DSPrompt**, respectively. So, user Dennis is authenticated by the Device Server and then chooses to configure a **Free Session** to the HR_Server using the Telnet protocol (Dennis could have attempted to access any host on the network).



Connecting Through the Device Server

When a user connects through the Device Server, that user is authenticated and is usually set up with a **User Service** that, once authentication is completed successfully, passes the user onto the specified host. Therefore, the **Line Service** is set to **DSLogin** and the **User Service** is set to whatever protocol the user will use to access the host; in this example, the **User Service** is set to **Dir Telnet**. When **User Service Dir Telnet** is selected, the IP address of the HR_Server is specified as the target Host IP. User Dennis will always have to log into the same server with this configuration.



Setting Up Lines

Line and port is often used interchangeably. They are almost the same, that is, each line has an associated port number (Line 1 is port 10001 by default).

How you set up a line is really determined by the device that is connected to the line. This section goes over some of the common ways a line is used and things that you will want to keep in mind when configuring the line.

DSLogin

When you configure the **Line** for **DSLogin**, users connecting to the Device Server will have to go through some form of authentication, either local or remote authentication. Regardless of whether a user has been configured in the User table (local authentication) or is inheriting the Default User's attributes (remote or Guest authentication), when a **User Service** is selected (other than **DSPrompt**), that connection (**Telnet**) will inherit the connection settings defined for **DSLogin**.

Direct/Silent/Reverse Connections

Direct connections bypass the Device Server, enabling the user to log straight into a specific host. A direct connection is recommended where a user logging in to the Device Server is not required. It is also recommended where multiple sessions are not a requirement. Direct connections require user interaction: the message **Press return to continue** is displayed on the user's screen and the session to the host is not initiated until **Enter** is pressed, after which the host login prompt is displayed. The message is redisplayed on logout.

Silent connections are the same as direct connections except that they are permanently established. The host login prompt is displayed on the screen. Logging out redispays this prompt. Silent connections, unlike direct connections, however, make permanent use of pseudo tty resources and therefore consume host resources even when not in use.

Reverse connections enable a host on the network to establish a connection through the Device Server port to a serial device.

Virtual Modems

Vmodem is a feature of the Device Server that provides “modem like” communication between two Device Servers on a network or between a Device Server and a host. This feature behaves like two modems connected across a telephone line. Typically, you use the **Vmodem** feature when you have multiple devices communicating with a central site. With just a single IOLAN Device Server at each end of the network, you don't need to use multiple modems, avoiding the associated costs of calls and connections.

The data is sent in raw format from the virtual modem and can be received by another Device Server or a host. This data can be sent automatically using the **Monitor DSR** option and then configuring the host and port number of the receiver; if the receiving side is also a Device Server, set the **Line Service** to **Rev Raw** or **Vmodem (Rev Raw** if the Device Server is only receiving, **Vmodem** to initiate bidirectional data flow) and the Device Server port that the data is coming in on (this should match the port number on the sending Device Server). Or, you can manually start a connection by typing **ATD<ip_address>,<port_number>** and end the connection by typing **+++ATH**. The **ip_address** can be in IPv4 or IPv6 formats and is the IP address of the receiver. For example, **ATD123.34.23.43,10001** or you can use **ATD12334234310001**, without any punctuation.

BIDIR

When you configure **BIDIR**, you are creating a bidirectional raw connection, meaning that the connection can be initiated from either the ethernet or serial side.

TruePort

When you configure a line for **TruePort**, the Device Server provides a complete COM port interface between the attached serial device and the network. You can also set the **Client Initiated** option, which allows either the client or the Device Server to initiate communication. See [TruePort on page 205](#) and the TruePort documentation for your operating system more information.

Signal I/O

When you configure a line for **Signal I/O**, you are using the DSR, DCD, CTS, DTR, and RTS serial pins for I/O channel digital input (DSR, DCD, and CTS) or digital output (DTR and RTS). Only after **Signal I/O** is specified for the **Line Service** can you configure the serial pins for I/O.

UDP

When you configure **UDP**, you are setting up a range of IP addresses and a port number that you will use to send UDP data to or receive UDP data from. For example:

The image shows a 'UDP Settings' dialog box with four entries. Each entry has fields for 'Start IP Address', 'End IP Address', 'Port', and 'Direction'. Entry 1 has IP range 172.16.1.1 to 172.16.1.25, Port 33001, and Direction 'Both'. Entry 2 has IP range 172.16.1.20 to 172.16.1.50, Port 33010, and Direction 'In'. Entry 3 has IP range 172.16.1.75 to 172.16.1.80, Port 33009, and Direction 'Out'. Entry 4 has IP range 0.0.0.0 to 0.0.0.0, Port 0, and Direction 'None'. There are 'OK' and 'Cancel' buttons at the bottom.

Entry	Start IP Address	End IP Address	Port	Direction
UDP Entry 1	172.16.1.1	172.16.1.25	33001	Both
UDP Entry 2	172.16.1.20	172.16.1.50	33010	In
UDP Entry 3	172.16.1.75	172.16.1.80	33009	Out
UDP Entry 4	0.0.0.0	0.0.0.0	0	None

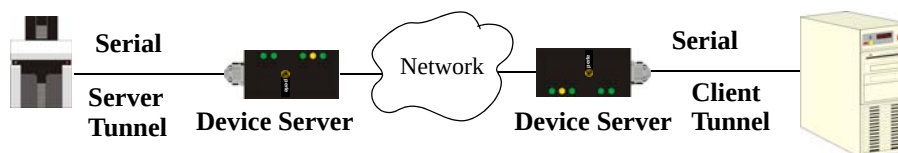
The UDP configuration window, taken from the DeviceManager, is configured to:

- UDP Entry 1**
 All hosts that have an IP address that falls within the range of **172.16.1.1** to **172.16.1.25** and listen to **Port 33001** will receive UDP data from the serial device. The serial device will only receive UDP data from the hosts in that range with a source **Port** of **33001**.
- UDP Entry 2**
 All UDP data received from hosts that have an IP address that falls within the range of **172.16.1.20** to **172.16.1.50** and **Port 33010** will be sent to the serial device. The Device Server will not send any data received on its serial port.
- UDP Entry 3**
 All hosts that have an IP Address that falls within the range of **172.16.1.75** to **172.16.1.80** and who listen to **Port 33009** will receive UDP data from the serial device. No UDP data will be sent to the serial device.
- UDP Entry 4**
 This entry is disabled since **Direction** is set to **None**.

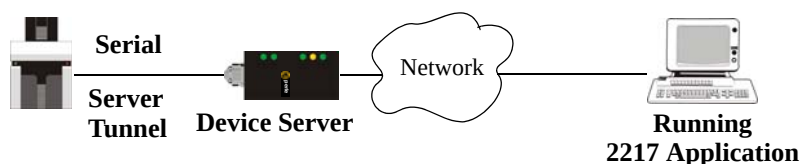
If **Direction** is set to **In** or **Both**, and **Port** is set to **0** (zero), the Device Server will learn a host IP Address and Port number based on the first UDP packet it receives and will then only send and/or accept UDP data from that host.

Serial Tunnel Settings

The purpose of the serial **Line Service Client/Server Tunnel** is to allow two Device Servers that are connected back-to-back over Ethernet to virtually link two serial ports, based on RFC 2217. The serial device that initiates the connection is the **Client Tunnel** and the recipient is the **Server Tunnel**, although once the serial communication tunnel has been successfully established, the tunnel will stay connected and communication can go both ways.



The **Server Tunnel** will also support Telnet Com Port Control protocol as detailed in RFC 2217.



The port signals will also follow the signals on the other port. If one port receives DSR then it will raise DTR on the other serial port. If one port receives CTS then it will raise RTS on the other port.

Setting Up Users

You can create up to four users, in addition to the Admin user (who cannot be deleted).

A user can even represent a device, like a barcode or a card swipe device, that you want to be authenticated.

User Accounts

When a serial device (like a dumb terminal or a barcode reader) is trying to access a host through the Device Server, you will need to configure user accounts when users:

- are authenticated by the Device Server and then connect to a host.
- want a single or multiple session(s) on a host; here they initially login to the Device Server before starting that session. The Device Server is used to configure and start the session.

When a host is accessing a serial device (like a modem or a server), you will need to configure user accounts where users:

- are using a reverse telnet connection to manage a UNIX server or a router.

User Levels

There are four **User Levels: Admin, Normal, Restricted, and Menu**. Setting up users is only necessary when the users are actually connecting to the Device Server. Oftentimes, the Device Server is used as a gateway to a network and the user never actually logs into the Device Server itself. Users who do log into the Device Server (**Line Service** set to **DSLogin** and **User Service** set to **DSPrompt**) will have to navigate by either the Menu or CLI (except for users with **Menu** privileges, who can only use the Menu).

- **Admin**—Users with **Admin** privileges have full administrative access to the IOLAN Device Server. This is not the same as the Admin user, but has equal authority (the Admin user is a permanent, factory-set user on the IOLAN Device Server).
- **Normal**—Users with **Normal** privileges have access to the Sessions menu and associated CLI only. They can start sessions, define and predefine sessions, and can change their own user environment.
- **Restricted**—Users with **Restricted** privileges have access to a restricted Sessions menu and associated CLI; they can only open sessions predefined for them by the Admin user, but not alter their own environment or sessions. Predefined sessions can also be configured to start automatically at login.
- **Menu**—Users with **Menu** privileges have access to predefined session. All other functionality is unavailable.

When the Admin user logs into the Device Server, the prompt ends with a #, whereas all other users' prompts ends with a \$ or £, depending on the character set.

Sessions

Sessions are defined for users who are coming in through a serial device going to a host on the LAN. Users who have successfully logged into the Device Server (**User Service** set to **DSPrompt**) can start up to four login sessions on LAN hosts. These users start sessions through the Menu option **Sessions**.

Multiple sessions can be run simultaneously on the same host or on different hosts. Users can switch between different sessions and also between sessions and the Device Server using hotkey commands.

Users with **Admin** or **Normal** privileges can define new sessions and connect through them, even configure them to start automatically on login to the Device Server. **Restricted** and **Menu** users can only start sessions predefined for them by the Admin user.

You can configure the User access rights to the port, such as **Read/Write (RW)** or **Read Input (RI)**.

Users From LAN to Device Server to Serial Device

Easy Port Access Menu

The Easy Port Access Menu is displayed when a **Restricted** or **Menu** level user logs into the box from the Ethernet side (**Line Service** set to **Rev Telnet** or **Rev SSH**) to access a serial device. The Easy Port Access Menu displays the line number, line name, line protocol (either **rev-tel** or **rev-ssh**), and a logout option. You can only access a line if it has the same connection protocol as the one you used to log into the Device Server. So, if you used SSH to log into the Device Server and the **Line Service** is set for **Rev Telnet**, you will not be able to access the serial device connected to that line.

Reverse Sessions and Multisessions

(2 Port+ only) The interaction between reverse sessions and multisessions is somewhat complex, so the definition of each parameter is provided to give you a context for their interaction.

Reverse Session—The definition of a Reverse Session is a TCP connection to a TCP port (defined in the Line configuration as the **DS Port**) on a line that is configured for **Reverse Telnet**, **Reverse SSH**, or **Reverse Raw**. Other LAN to Device Server connections, such as the connection to the well-known Telnet or SSH ports (the management TCP ports of 23 for Telnet and 22 for SSH), do not count as reverse sessions.

Reverse Session Limit—This server parameter is the maximum number of **Reverse Sessions** (as defined above) that can be active on the Device Server at any given moment. This value is completely independent of the Line **Multisessions** number. If the **Reverse Session Limit** is set to a maximum of 20, that only says that the total number of reverse sessions cannot exceed 20; it says nothing about total number of reverse sessions per line, total number of active line, etc.

The minimum value is **1**. The maximum value is determined as follows:

1. Add up all the serial ports, internal modems, and PCI slots in the Device Server. This number equals the maximum number of lines available on the Device Server.
2. If the maximum number of lines is less than 4 exclusive (that is, 1, 2, or 3), then multiply that number by 4 to obtain the maximum Server **Reverse Session Limit**, otherwise, multiply the maximum number of lines by 2 to obtain the maximum Server **Reverse Session Limit**.

Multisessions—The Line **Multisessions** limit is the number of additional reverse sessions (beyond the first reverse session) allowed on the line. Therefore, if this number is set to 5, the total number of reverse sessions allowed on the line is 5+1 or 6.

The default and minimum value is **0**, which means that reverse **Multisessions** is disabled and you only get the 1 default reverse session; therefore, the maximum value is the maximum **Server Reverse Session Limit**-1.

Reverse Session Security—If this Line parameter is enabled, the user must login when making a **Reverse Telnet** connection to a configured **DS Port** (the TCP port associated with the line that is configured for a reverse connection). If the line is configured for **Reverse SSH**, this option is redundant, because a login is always required when SSH is specified.

So, in order to make multiple reverse connections to a line, the line must be configured to log users in so the user's Line Access Rights can be obtained from the User profile (regardless of whether that profile comes from a defined User, the Default User's settings, or is passed to the Device Server from TACACS+ or RADIUS). This means that one of the following conditions have been met:

- The line is configured for **Reverse Telnet** and **Reverse Session Security** is enabled.
- The line is configured for **Reverse SSH**.

If users are connecting to the Device Server by the management TCP port (or well-known port of 23 for Telnet and 22 for SSH), they always have to login to the Device Server and depending on their access level can get:

admin—standard CLI/Menu with admin privileges

normal—standard CLI/Menu with normal privileges

restricted—Easy Port Configuration Menu

menu—Easy Port Configuration Menu

If users are connecting to the Device Server by the IP address and configured **DS Port** (for example, the line has a configured **DS Port** of **88**, so the user would connect with the **telnet 101.170.12.15 88** command), a line set up for **Reverse Telnet** would require a login if **Reverse Session Security** is enabled, otherwise, the User settings would dictate what the user sees. For **Reverse SSH**, the user always has to login to the Device Server.

Configuring Network Options

Hosts

This is probably one of the first Device Server options you want to configure, since so many other configuration options require a preconfigured host. You can use any host name you want, since the host name is used only by the Device Server. You can configure up to 20 hosts using IPv4 or IPv6 internet addresses.

Gateways

Gateways are hosts that connect Local Area Networks (LANs) together. If you want to access a host that isn't on your local network, you will be connected via a gateway. Gateways route data via other gateways until the destination local network is reached. There are three types of gateways:

- **Default**—A gateway that provides general access beyond your local network.
- **Host**—A gateway reserved for accessing a specific host external to your local network.
- **Network**—A gateway reserved for accessing a specific network external to your local network.

You can specify up to twenty gateways.

Syslog

The system log is sent to the specified host. You can configure a primary and secondary host for the syslog information and specify the level for which you want syslog information sent.

SNMP

If you are using SNMP to manage/configure the Device Server, or to view statistics or traps, you must set up a User in SNMP version 3 or a Community in SNMP version 1,2 to allow your SNMP manager to connect to the Device Server; this can be done in the DeviceManager, WebManager, CLI, or Menu. You must then load the perle-ds1.MIB (found on the CD-ROM packaged with the Device Server) file into your SNMP manager before you connect to the Device Server.

Configuring Time

The Device Server has an internal clock that can be set, but it will be reset during a reboot or a power outage.

Language support

Two language files, in addition to English, are supplied on the supplemental CD, French and German. You can use any of these language files to create a translation into a language of your choice. You can download the language file (whether the language is supplied or translated) into the Device Server and select the **Language** option of **Customlang** (custom language), making the Menu, CLI, and WebManager field labels display in your language.

You can view Menu, CLI, or WebManager in one other language only (as well as English). If you download another language file, this new language will replace the first language you downloaded.

You can revert to English at any time; the English language is stored permanently in the Device Server and is not overwritten by your new language. Each user logged into the Device Server can operate in either English or the downloaded language.

Loading a Supplied Language

This section describes how to download a language file using the CLI, since it is the least intuitive method. French and German language files are provided on the supplemental CD.

To load one of the supplied languages into the Device Server, so the Menu, CLI and WebManager fields appear in another language, do the following:

1. Open the supplemental CD and identify the language file, either **Iolan_ds_French.txt** or **Iolan_ds_German.txt**, or supply one of your own translated files.
2. Copy the language file to a host machine on the network; place it in the main file system or on the main hard drive.
3. Either use the TFTP defaults in the Device Server or, configure as necessary, TFTP in the Device Server.
4. In the CLI of the Device Server, enter the host IP address and file name; for example,

```
netload customlang 172.16.4.1 /temp/Iolan_ds_French.txt
```

Do *not* enter a drive letter! Also, the path and/or file name must begin with the forward slash (/) character.

The Device Server will download the language file via TFTP.

5. To set an individual user to the new language, go to the **Users** menu and, in the **Language** field select **Customlang**. In the CLI (only) you can set individual users or all users to the new language; see the **set user *** command.
6. The user will see the change of language when he/she logs out (**Main Menu**, **Sessions Menu**, **Logout**) and logs back into the Device Server. If, as Admin user, you change your language setting to **Customlang**, you will see the text menus display in the new language when you save and exit the **Change User** form.

Note: If you download a new software version, you can continue to use your language unchanged; however, we recommend translating the new strings, which will be added to the end of the language file. A **Reset to Factory Defaults** will reload the **Customlang** as English.

On successful download, the **Customlang** in the Device Server will be overwritten by the new language.

Translation Guidance

To help you with your translation, of supplied ASCII text language files we offer the following guidance:

- The Device Server will support languages other than English (and the supplied German and French languages). The English language file, **english.txt**, displays the character length of each line at the beginning of the line. If a translated line goes over that character length, it will be displayed truncated in the Menu, CLI, or WebManager.
- Translate line for line, do not omit lines if you do not know the translation; leave the original untranslated text in place. Also, you must maintain the same sequential order of lines. It is a good practice to translate the file using a text editor that displays line numbers, so you can periodically verify that the line sequence has not changed from the original file (by comparing it to the original file).
- Keep all translations in quotes, otherwise the line will not display properly.
- Each line must end with a carriage return.
- If a line contains only numbers, for example 38400, leave that line in place, unchanged (unless you are using a different alphabet).

Software Upgrades and Language Files

If you receive a software upgrade for the Device Server, the language files supplied on the supplemental diskette/CD might also have been updated. We will endeavour to provide a list of those changes in another text file on the same supplemental CD.

Note: The upgrade of your software (firmware) will not change the display of the language in the Menu, CLI, or WebManager.

If you are already using one of the supplied languages, French or German, you probably want to update the language file in the Device Server. Until you update the Device Server with the new language file, new text strings will appear in English.

If you are already using a language translated from an earlier version, you probably want to amend your translation. When a language file is updated, we will try to maintain the following convention:

1. New text strings will be added to the bottom of the file (not inserted into the body of the existing file).
2. Existing text strings, if altered, will be altered in sequence; that is, in their current position in the file.
3. The existing sequence of lines will be unchanged.
4. Until you have the changes translated, new text strings will appear in the Menu, CLI, or WebManager in English.

Downloading Terminal Definitions

All terminal types can be used on the Device Server. Some terminal types which are not already defined in the Device Server, however, are unable to use Full Screen mode (menus) and may not be able to page through sessions properly. When installed, the Device Server has several defined terminal types—Dumb, WYSE60, VT100, ANSI, TVI925, IBM3151, VT320, and HP700.

If you are not using, or cannot emulate, any of these terminal types, you can add up to three additional terminal definitions to the Device Server. The terminal definitions can be downloaded from a TCP/IP host.

To download terminal definitions, follow these steps:

1. Decide which TCP/IP host you are going to use. It must be a machine with TFTP enabled.
2. Configure TFTP in the Device Server as necessary.
3. Download the new terminal definition to the Device Server as **Term1**, **Term2**, or **Term3**.
4. In the **Line** configuration, select the **Terminal Type Termx** that you custom defined.

Creating Terminal Definition Files

To create new terminal definition files, you need to copy and edit the information from the terminfo database.

1. On a UNIX host, change directory to **/usr/lib/terminfo/x** (where **x** is the first letter of the required terminal type). For a Wyse60, for example, you would enter the command **cd /usr/lib/terminfo/w**.
2. The termcap files are compiled, so use the command **infocmp termfile** to read the required file (for example: **infocmp wy60**).
3. Check the file for the attribute **xmc#n** (where **n** is greater than or equal to 1). This attribute will corrupt menu and form displays making the terminal type unsuitable for using Menu mode.
4. If the terminal definition is suitable, change to a directory of your choice.

5. Rename and copy the file to the directory specified at step 4. using the command **infocmp termfile > termn** where *n* is greater than or equal to 1; (for example, **infocmp wy50 > term1**). Make sure the file has global read and execute permission for its entire path.
6. Edit the file to include the following capabilities in this format:

```
term=
acsc=
bold=
civis=
clear=
cnorm=
cup=
rev=
rmacs=
rmso=
smacs=
smso=
page=
circ=
```

For example:

```
term=AT386 | at386| 386AT |386at |at/386 console
acsc=jYk?lZm@qDtCu4x3
bold=\E[1m
civis=
clear=\E[2J\E[H
cnorm=
cup=\E[%i%p1%02d;%p2%02dH
rev=\E4A
rmacs=\E[10m
rmso=\E[m
smacs=\E[12m
smso=\E[7m
page=
circ=n
```

Note: As you can see from the example, capabilities which are not defined in the terminfo file must still be included (albeit with no value). Each entry has an 80 character limit.

On some versions of UNIX, some of the capabilities are appended with a millisecond delay (of the form \$<n>). These are ignored by the Device Server and can be left out.

The 'acsc' capability, if defined, contains a list of character pairs. These pairs map the characters used by the terminal for graphics characters to those of the standard (VT100) character set.

Include only the following character pairs:

jx, kx, lx, mx, qx, tx, ux and xx

(where *x* must be substituted by the character used by the terminal). These are the box-drawing characters used to display the forms and menus of Menu mode. They must be entered in this order.

The last two capabilities will not be found in the terminfo file. In the **page** field you must enter the escape sequence used by the terminal to change screens. The **circ** field defines whether the terminal can use **previous page** and **next page** control sequences. It must be set to **y** or **n**. These capabilities can be found in the documentation supplied with the terminal.

TFTP Configuration

TFTP can be configured for two unique transfer operations:

1. Between the DeviceManager and a Device Server. This configuration is accessed by selecting Tools, Options from the DeviceManager's tool bar. You can specify the number of times the DeviceManager's TFTP server retries a file transfer to a Device Server, how long the TFTP process will wait (timeout) before retrying to transfer a file, and the UDP port that will be used for the file transfer between the DeviceManager and the Device Server. (DeviceManager only.)
2. Between the Device Server and a host. This configuration is accessed by selecting Network, TFTP in the DeviceManager, by typing **set server tftp** in the CLI, by selecting **Network Configuration, TFTP** from the Menu, by selecting **ServerInfo, tftpRetry** and **tftpTimeOut** in the SNMP MIB, or by selecting **Network, TFTP** in the WebManager. You can configure the number of times the Device Server's TFTP client retries a file transfer to a host and how long the TFTP process will wait (timeout) before retrying to transfer a file.

You must have a TFTP server running on any host that you are uploading or downloading files to/from. If you are using the DeviceManager and transferring a local file to a Device Server, you still need to have a TFTP server running on your PC. When you specify the file path, the path must be relative to the default path set in your TFTP server software (do not use drive letters in the file path).

Resetting Configuration Parameters

You can reset the Device Server to its factory settings through any of the following methods:

- You can push in the recessed button at the back of the Device Server hardware for more than three seconds (pushing it in and then quickly releasing will just reboot the Device Server)
- DeviceManager, select **Tools, Reset to Factory Defaults**
- CLI, at the command line type, **reset factory**
- WebManager, click the **Factory Defaults** button
- Menu, select **Network Configuration, Reset to Factory Defaults**
- SNMP, in the **adminInfo** folder, **Set** the **adminFunction** variable to **2**

Lost Admin Password

If the Admin user password is lost, there are only two possible ways to recover it:

- reset the Device Server to the factory defaults
- have another user that has **admin** level rights, if one is already configured, reset the Admin password

DHCP/BOOTP

You can use DHCP/BOOTP to perform the following actions on a single or multiple Device Servers on bootup:

- auto-configure with minimal information; for example, only an IP address
- auto-configure with basic setup information (IP address, subnet/prefix bits, etc.)
- download a new version of firmware
- download a full configuration file

DHCP/BOOTP is particularly useful for multiple installations: you can do all the Device Server's configuration in one DHCP/BOOTP file, rather than configure each Device Server manually. Another advantage of DHCP/BOOTP is that you can connect a Device Server to the network, turn on its power and let autoconfiguration take place. All the configuration is carried out for you during the DHCP/BOOTP process.

DHCP/BOOTP Parameters

The following parameters can be set in the DHCP/BOOTP bootp file:

- **SW_FILE**—The full path, pre-fixed by hostname/IP address (IPv4 or IPv6), and file name of the firmware update.
- **CONFIG_FILE**—The full path, pre-fixed by hostname/IP address (IPv4 or IPv6), and file name of the configuration file. Note: these parameters include clear text user passwords.
- **GUI_ACCESS**—Access to the Device Server from the HTTP WebManager. Values are **on** or **off**.
- **SECURITY**—Restricts Device Server access to devices listed in the Device Server's host table. Values are **yes** or **no**.
- **TFTP_RETRY**—The number of TFTP attempts before aborting. This is a numeric value, for example, 5.
- **TFTP_TMOUT**—The time, in seconds, before retrying a TFTP download/upload. This is a numeric value, for example, 3.
- **CUSTOM_LANG**—The full path, pre-fixed by a hostname/IP address (IPv4 or IPv6), and file name of a translated language file. For example, 192.101.34.211 /accounting/Iolan_ds_german.txt.
- **EXTRA_TERM1**—(**EXTRA_TERM2**, **EXTRA_TERM3**) The full path, pre-fixed by a hostname/IP address (IPv4 or IPv6), and file name of a termcap file for a specific terminal type.

Creating Custom Applications

You can create custom applications for the Device Server by using the Perle SDK. See the *SDK Programmer's Guide* (the SDK and guide are found on the Perle website at www.perle.com/downloads/index.shtml) for information about the functions that are supported. You must download the program and any ancillary files to the Device Server and set the **Line Service** to **Custom App** to run a custom application. You must also specify the program executable in the **Program Command Line** parameter.

I/O Model Features

There is a line of I/O Device Servers that can control/monitor the following types of I/O:

- Analog (Input)
- Digital Input/Output
- Relay
- Temperature

Some of the models are I/O combinations and some of the models support one I/O type.

Failsafe Timer

The Failsafe Timer is enabled on a global basis and provides a trigger mechanism that can be configured for each channel when no I/O traffic/management has occurred for the specified amount of time. A Failsafe Action can be configured for each Digital Output channel, each Serial Signal Output channel (DTR and RTS), and each Relay channel to either Activate or Deactivate the output.

Alarms

Analog and Temperature input models support an Alarm mechanism in which you can specify up to five severity levels of alarm triggers and clear levels; the alarm triggers/clear levels can activate in either increasing or decreasing severity levels.

The Digital input supports an Alarm mechanism based on a trigger of either active input or inactive input and can be cleared either manually or automatically (when the trigger condition goes inactive or active, respectively).

Each time an alarm is triggered or cleared, you can specify any combination of the following to be initiated:

- An SNMP trap
- A message to syslog

UDP

The I/O UDP broadcast feature periodically broadcasts the I/O channel status in a UDP message.

You can configure up to four sets of IP address entries (each entry consisting of a start and end IP address range) to broadcast I/O status data. The data depends on the I/O model (Analog, Digital, Serial Signals) and contains information for all channels.

UDP Unicast Format

In order to interpret the UDP unicast data, you must use the following tables to decipher the appropriate data. If your model does not support a data format (for example, digital data) or you do not have any channels configured for a data format, it will be included in UDP broadcast package, with a Total Length of 0 (zero) and no data following.

Version	Total Length	Analog Data	Digital Data	Serial Signal Data
---------	--------------	-------------	--------------	--------------------

Each section, with the exceptions of the Version and Total Length sections, is comprised of its own subset of bytes.

Analog Data

Each Analog channel is comprised of the following data fields (big Endian format):

Total Length	*Data Exists	Data					
2 Bytes	1 Byte (in bits)	curRawValue 2 Bytes	minRawValue 2 Bytes	maxRawValue 2 Bytes	curEngValue 4 Bytes	minEngValue 4 Bytes	maxEngValue 4 Bytes

The following section describes the values in the Analog Data field:

- **curRawValue**—The current raw value received from the Analog to Digital converter.
- **minRawValue**—The minimum value received from the Analog to Digital converter until it is cleared.
- **maxRawValue**—The maximum value received from the Analog to Digital converter until it is cleared.
- **curEngValue**—The current converted value (voltage/current for Analog or Celsius/Fahrenheit for Temperature).
- **minEngValue**—The minimum converted value (voltage/current for Analog or Celsius/Fahrenheit for Temperature) until it is cleared.
- **maxEngValue**—The maximum converted value (voltage/current for Analog or Celsius/Fahrenheit for Temperature) until it is cleared.

Digital/Relay Data

The digital data is in bit format, 1 meaning On and 0 (zero) meaning Off. Each channel has its own bit, in least significant bit order.

Length	*Data Exists	Data (1 Byte, one bit for each channel)							
2 Bytes	1 Byte (in bits)					Channel4	Channel3	Channel 2	Channel 1

Serial Signal Data

The serial data is in bit format, 1 meaning On and 0 (zero) meaning Off. Each channel has its own bit, in the following order.

Length	*Data Exists	Data (1 Byte for each port, one bit for each signal)							
2 Bytes	1 Byte (in bits)				RTS	DTR	CTS	DCD	DSR

*Data Exits Field

The Data Exists field is 1 byte in least significant bit order, for each channel. If data exists for a channel, the bit will be 1, if no data exists for a channel (it is not configured), the bit will be 0 (zero).

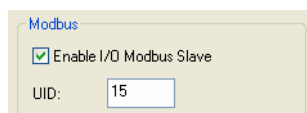
Data Exists(1 Byte, one bit for each channel)							
				Channel4	Channel3	Channel 2	Channel 1

UDP Unicast Example

For an example of the I/O UDP unicast, see the sample program, **ioudpbcast.c**, found on your CD-ROM.

I/O Modbus Slave

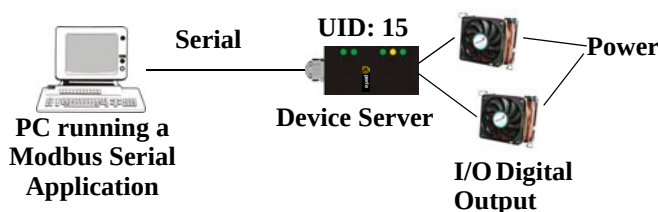
If you have a Modbus serial or TCP application, it can access I/O connected to the Device Server when the I/O Global Modbus Slave is enabled. You must supply a unique UID for the Device Server, as it will act as a Modbus Slave.



There are three ways your Modbus Application can connect to the Device Server to access I/O.

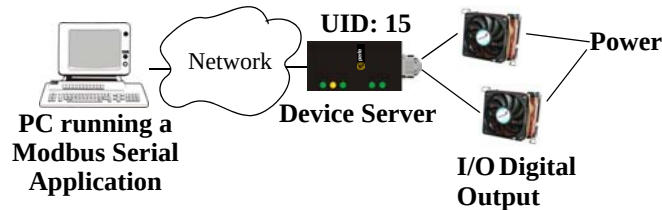
Modbus Serial Application Connected to the Serial Port

Your Modbus serial application can be connected right to the Device Server serial port to access I/O.



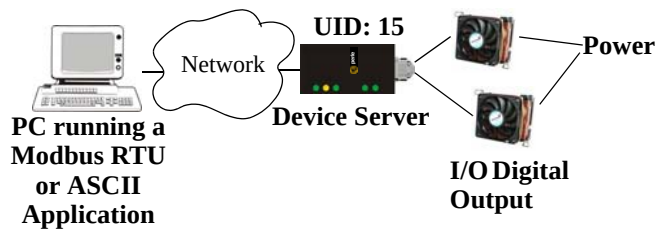
Modbus Serial Application Connected to the Network

If you want to access the I/O from a LAN connection, you can install TruePort on the PC running the Modbus serial application as described in [TruePort on page 82](#) and connect to the Device Server over the network.



Modbus TCP Application

If you have a Modbus RTU or Modbus ASCII program, you can access the I/O by connecting to the Device Server over the network.



Modbus I/O Access

The section defines the function codes and registers you will need to access the I/O through Modbus TCP, Modbus serial, or Modbus serial/TruePort.

Function Codes

The following function codes are supported by the Device Server:

- 01 read coils
- 03 read multiple holding registers
- 04 read input registers
- 05 write coil
- 06 write single register
- 08 diagnostics (echo the request)
- 15 force multiple coils
- 16 write multiple registers

There are four Modbus data models:

Discrete Input	Not used
Coils	Digital Input (DI), Alarm state for DI, Digital Output (DO)
Input Registers (IR)	Analog Input (AI), Alarm state for AI
Holding Registers	Status (R), Control value (R/W or W)

All coil/register values are in decimal.

I/O Coil/Register Descriptions

This section contains descriptions of I/O coils:

- **MB_REG_DI_SENSOR**—Status of Digital input. 1 is Active, 0 is Inactive. If **Invert Signal** is configured **on**, 0 is Active, 1 is Inactive. If input is **Latched**, returns latched status.
- **MB_REG_DI_SENSOR_ALARM_STATE**—Indication if input is in alarm state. 1 is In Alarm state, 0 is Not in Alarm state. A write of any value clears the alarm state.
- **MB_REG_DO_SENSOR**—Status of Digital output. 1 is Active, 0 is Inactive. If **Invert Signal** is configured **on**, 0 is Active, 1 is Inactive.

This section contains descriptions of I/O holding registers:

- **MB_REG_HR_DI_SENSOR_LATCH**—The latch status of the Digital input. 1 is Latched, 0 is Not latched. A write of any value will clear the latch.
- **MB_REG_HR_DO_SENSOR_PULSE_ISW**—Inactive Signal Width. This is how long the channel will remain inactive during pulse mode in increments of 100ms. Valid values are 1-9999. The default is 1 (100 ms).
- **MB_REG_HR_DO_SENSOR_PULSE_ASW**—Active Signal Width. This is how long the channel will be active during the pulse mode in increments of 100ms. Valid values are 1-9999. The default is 1 (100 ms).
- **MB_REG_HR_DO_SENSOR_PULSE_COUNT**—The number of times the channel output will pulse. Each count consists of an active/inactive sequence. The default is 1 cycle.
- **MB_REG_HR_AI_CLEAR_ALARM_LATCH**—Used to reset a latched alarm state. A write of any value will clear the alarm latch for the specific Analog input.
- **MB_REG_HR_AI_CLEAR_MAX**—Used to reset the Analog input maximum value reached. A write of any value will reset the maximum.
- **MB_REG_HR_AI_CLEAR_MIN**—Used to reset the Analog input minimum value reached. A write of any value will reset the minimum.

This section contains descriptions of I/O input registers:

- **MB_REG_IR_CURR_ENG**—The current value of an Analog or Temperature input converted to appropriate units. For Analog, this will be in voltage or current, depending on the configuration. For the Temperature, this value will be in Celsius or Fahrenheit, depending on configuration.
- **MB_REG_IR_MIN_ENG**—The minimum converted value ever reached on this input since the Device Server was re-started or a manual clear was issued.
- **MB_REG_IR_MAX_ENG**—The maximum converted value ever reached on this input since the Device Server was re-started or a manual clear was issued.
- **MB_REG_IR_CURR_RAW**—The current raw value received from the Analog to Digital converter. This is a hexadecimal value in the range of 0 -0xFFFF.
- **MB_REG_IR_MIN_RAW**—The minimum raw value ever reached on this input since the Device Server was re-started or a manual clear was issued.
- **MB_REG_IR_MAX_RAW**—The maximum converted value ever reached on this input since the Device Server was re-started or a manual clear was issued.
- **MB_REG_IR_ALARM_LEVEL**—This gives the current alarm severity level for the corresponding Analog input. Severity levels range from 0 (not in alarm) to 5 (highest alarm severity).

Serial Port Coil/Register Descriptions

This section contains descriptions of serial port coils:

- **MB_REG_DI_DSR**—The status of the DSR input signal. 1 is Active, 0 is Inactive. If **Invert Signal** is configured **on**, 0 is Active, 1 is Inactive. If input is **Latched**, returns latched status.
- **MB_REG_DI_DSR_ALARM_STATE**—The alarm state of DSR input signal. 1 is In Alarm state, 0 is Not in Alarm state. A write of any value clears the alarm state.
- **MB_REG_DI_DCD**—The status of DCD line. 1 is Active, 0 is Inactive. If **Invert Signal** is configured **on**, 0 is Active, 1 is Inactive.
- **MB_REG_DI_DCD_ALARM_STATE**—The alarm state of DCD input signal. 1 is in Alarm state, 0 is Not in Alarm state. A write of any value clears the alarm state.
- **MB_REG_DI_CTS**—The status of CTS input signal. 1 is Active, 0 is Inactive. If **Invert Signal** is configured **on**, 0 is Active, 1 is Inactive.
- **MB_REG_DI_CTS_ALARM_STATE**—The alarm state of CTS input signal. 1 is Alarm, 0 is Not in Alarm. A write of any value clears the alarm state.
- **MB_REG_DO_DTR**—The status of DTR output signal. 1 is Active, 0 is Inactive.
- **MB_REG_DO_RTS**—The status of RTS output signal. 1 is Active, 0 is Inactive.

This section contains descriptions of serial port holding registers:

- **MB_REG_HR_DI_DSR_LATCH**—The latched status for the DSR signal. 1 is Latched, 0 is Not Latched. A write any value will clear the latch.
- **MB_REG_HR_DI_DCD_LATCH**—The latched status for the DCD signal. 1 is Latched, 0 is Not Latched. A write any value will clear the latch.
- **MB_REG_HR_DI_CTS_LATCH**—The latched status for the CTS signal. 1 is Latched, 0 is Not Latched. A write any value will clear the latch.

A4/T4 Registers

The following registers are supported by the Device Server A4 and T4 input models:

	Data Model	A1/T1	A2/T2	A3/T3	A4/T4	R/W
Holding Registers:						
	MB_REG_HR_AI_CLEAR_ALARM_LATCH	2049	2050	2051	2052	W
	MB_REG_HR_AI_CLEAR_MAX	2113	2114	2115	2116	W
	MB_REG_HR_AI_CLEAR_MIN	2177	2178	2179	2180	W
Input Registers:						
	MB_REG_IR_CURR_ENG	2080	2112	2144	2176	R
	MB_REG_IR_MIN_ENG	2082	2114	2146	2178	R
	MB_REG_IR_MAX_ENG	2084	2116	2148	2180	R
	MB_REG_IR_CURR_RAW	2086	2118	2150	2182	R
	MB_REG_IR_MIN_RAW	2087	2119	2151	2183	R
	MB_REG_IR_MAX_RAW	2088	2120	2152	2184	R
	MB_REG_IR_ALARM_LEVEL	2089	2121	2153	2185	R

A4D2/A4R2 Registers

The following coils and registers are supported by the Device Server A4D2 and A4R2 I/O models:

	Data Model	A1	A2	A3	A4	D1/R1	D2/R2	R/W
Coils:								
	MB_REG_DI_SENSOR	----	----	----	----	6149	6150	R
*	MB_REG_DI_SENSOR_ALARM_STATE	----	----	----	----	6213	6214	R/W
	MB_REG_DO_SENSOR	----	----	----	----	6661	6662	R/W
Holding Registers:								
	MB_REG_HR_DI_SENSOR_LATCH	----	----	----	----	6149	6150	R/W
	MB_REG_HR_DO_SENSOR_PULSE_ISW	----	----	----	----	6213	6214	R/W
	MB_REG_HR_DO_SENSOR_PULSE_ASW	----	----	----	----	6277	6278	R/W
	MB_REG_HR_DO_SENSOR_PULSE_COUNT	----	----	----	----	6341	6342	R/W
	MB_REG_HR_AI_CLEAR_ALARM_LATCH	2049	2050	2051	2052	----	----	W
	MB_REG_HR_AI_CLEAR_MAX	2113	2114	2115	2116	----	----	W
	MB_REG_HR_AI_CLEAR_MIN	2177	2178	2179	2180	----	----	W
Input Registers:								
	MB_REG_IR_CURR_ENG	2080	2112	2144	2176	----	----	R
	MB_REG_IR_MIN_ENG	2082	2114	2146	2178	----	----	R
	MB_REG_IR_MAX_ENG	2084	2116	2148	2180	----	----	R
	MB_REG_IR_CURR_RAW	2086	2118	2150	2182	----	----	R
	MB_REG_IR_MIN_RAW	2087	2119	2151	2183	----	----	R
	MB_REG_IR_MAX_RAW	2088	2120	2152	2184	----	----	R
	MB_REG_IR_ALARM_LEVEL	2089	2121	2153	2185	----	----	R

*For DI alarm state, read will get state, write will clear alarm.

D4/D2R2 Registers

The following coils and registers are supported by the Device Server D4 and D2R2 I/O models:

	Data Model	D1	D2	D3/R1	D4/R2	R/W
Coils:						
	MB_REG_DI_SENSOR	6145	6146	6147	6148	R
*	MB_REG_DI_SENSOR_ALARM_STATE	6209	6210	6211	6212	R/W
	MB_REG_DO_SENSOR	6657	6658	6659	6660	R/W
Holding Registers:						
	MB_REG_HR_DI_SENSOR_LATCH	6145	6146	6147	6148	R/W
	MB_REG_HR_DO_SENSOR_PULSE_ISW	6209	6210	6211	6212	R/W
	MB_REG_HR_DO_SENSOR_PULSE_ASW	6273	6274	6275	6276	R/W
	MB_REG_HR_DO_SENSOR_PULSE_COUNT	6337	6338	6339	6340	R/W

*For DI alarm state, read will get state, write will clear alarm.

Serial Signals

The following coils and registers are supported by the Device Server I/O models:

	Data Model	Pin	R/W
Coils:			
	MB_REG_DI_DSR	4225	R
	MB_REG_DI_DSR_ALARM_STATE	4289	R/W
	MB_REG_DI_DCD	4353	R
	MB_REG_DI_DCD_ALARM_STATE	4417	R/W
	MB_REG_DI_CTS	4481	R
	MB_REG_DI_CTS_ALARM_STATE	4545	R/W
	MB_REG_DO_DTR	4673	R/W
	MB_REG_DO_RTS	4737	R/W
Holding Registers:			
	MB_REG_HR_DI_DSR_LATCH	4097	R/W
	MB_REG_HR_DI_DCD_LATCH	4609	R/W
	MB_REG_HR_DI_CTS_LATCH	5121	R/W

TruePort

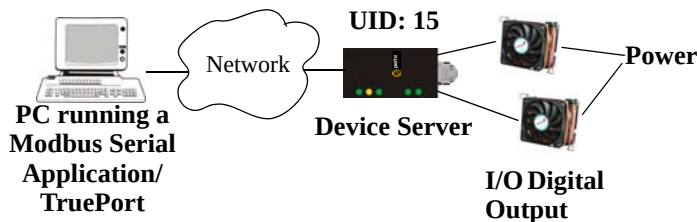
You can see a sample API I/O over TruePort program called **ioapiotp.c** on the CD-ROM.

TruePort/Modbus Combination

If you have a Modbus serial application running on a PC that is connected to a network, you can use TruePort as a virtual serial connection to communicate with the Device Server over the network to access I/O data. You also have the option of enabling SSL as a security option to encrypt the data that is communicated between the Device Server and the host machine (SSL/TLS must be configured in the Server settings and on the TruePort host).

The screenshot shows the TruePort configuration window with the following settings:

- Modbus:**
 - ☒ Enable I/O Modbus Slave
 - UID: 15
- TruePort:**
 - ☒ Enable I/O TruePort Services
 - ☐ Enable SSL Encryption
 - Listening Port: 33816 TCP
 - TruePort I/O Services:**
 - I/O Access via API: Enabled
 - Modbus Serial I/O Access: Enabled



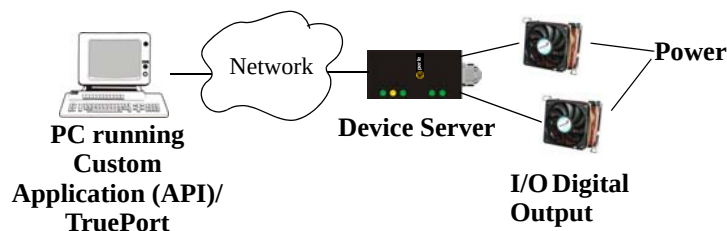
The host running TruePort must be in Modbus/ASCII or Modbus/RTU mode.

API Over TruePort Only

If you have a custom application that talks to a serial port, you can use TruePort as a virtual serial port to communicate with the Device Server over the network to access I/O data using the Perle API. You also have the option of enabling SSL as a security option to encrypt the data that is communicated between the Device Server and the host machine (SSL/TLS must be configured in the Server settings and on the TruePort host). See [API I/O Access Over TruePort on page 206](#) for more information on the API.)

The screenshot shows the TruePort configuration window with the following settings:

- TruePort:**
 - ☒ Enable I/O TruePort Services
 - ☐ Enable SSL Encryption
 - Listening Port: 33816 TCP
 - TruePort I/O Services:**
 - I/O Access via API: Enabled
 - Modbus Serial I/O Access: Disabled



The host running TruePort must be in I/O API mode.

Digital Channels

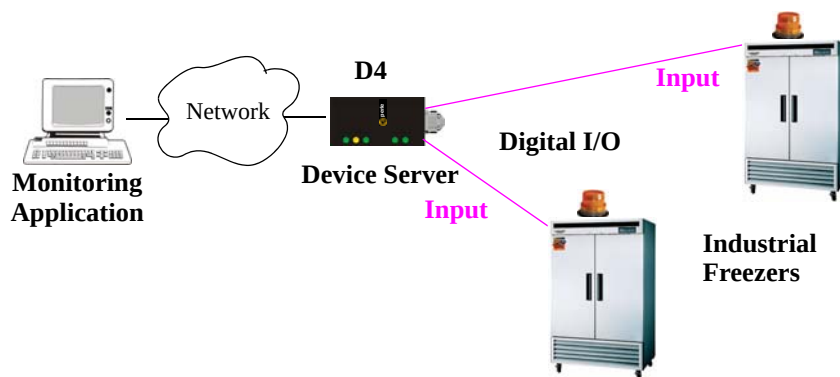
You can configure the Digital I/O channels for either input or output, making the Device Server flexible in your production environment. Jumpers must be set to correspond with the software setting of either Input or Output; see [Digital I/O Module on page 33](#) for jumper settings.

Digital Input

The Digital input channels allow you to configure the following options:

- You can choose to remember the last state change, or latch, that occurred. Your options are to latch (remember) when the state changes from inactive to active or active to inactive.
- You can choose to invert the signal, which is useful if your sensor is wired in such a way that closed is actually inactive, whereas closed is normally considered active.
- You can also configure an alarm trigger and clear mode based on whether the Digital input is active or inactive, sending an email, syslog message, and/or SNMP trap when the alarm is triggered or cleared.

In an industrial freezer warehouse example, a D4 is used to monitor the open door sensor, so that every time a freezer door is opened, an alarm is triggered and a syslog message is sent to syslog, where the monitoring application notes the time.

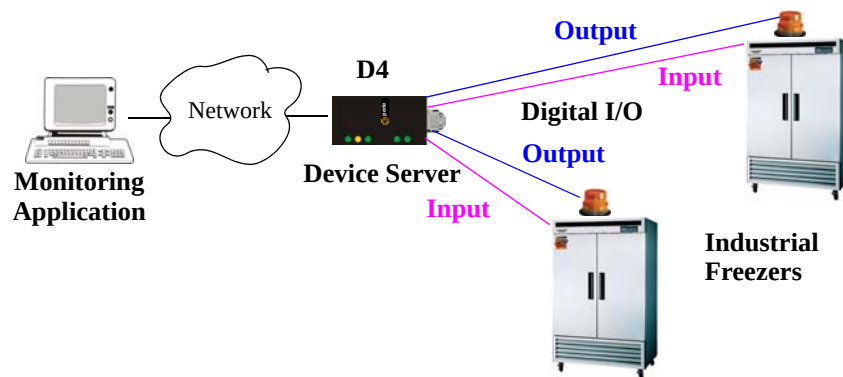


Digital Output

The Digital output channels support three types of Digital output: sink (voltage), source (ground), and sink and source (apply voltage or ground). For the output type, you can configure the following options:

- You can choose to manually activate/deactivate the Digital output.
- You can choose to manually activate/deactivate the Digital output and then specify that the Digital output will either pulse (you get to specify the active and inactive pulse times) continuously or for a specified number of pulse counts.
- You can choose to manually activate/deactivate the Digital output and then specify a delay before the output goes from inactive to active or active to inactive.
- You can also specify a failsafe action that can either activate or inactivate the Digital output when the failsafe timer is triggered (see [Failsafe Timer](#) on page 74 for more information).

When one of the industrial freezer doors are left open for more than five minutes, the Monitoring Application (using the Perle API) starts the Digital output sink, causing the strobe light on top of the offending freezer to activate.



Temperature Channels

Temperature input channels monitor RTD or thermocouple temperature sensors. You can also configure severity alarms that can send an email, a syslog message, and/or an SNMP when an alarm is triggered or cleared; See [Alarms on page 75](#) for more information about the alarms.

RTD ranges are:

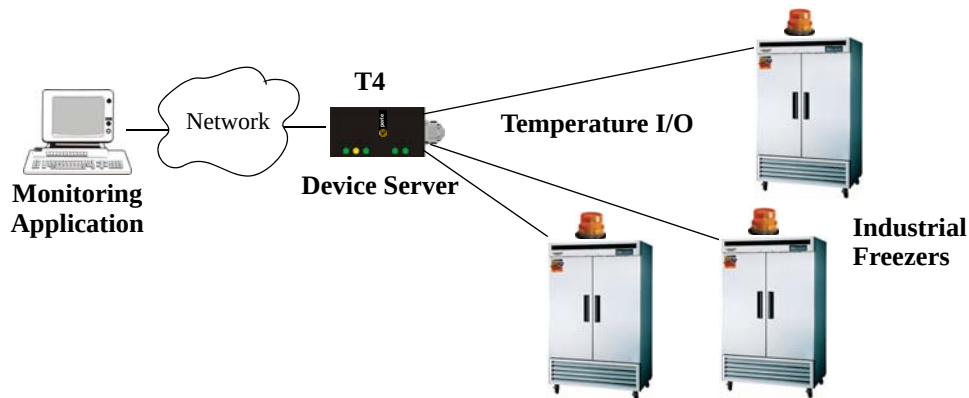
- Pt100 a=385 -50 to 150C ● Pt100 a=392 -50 to 150C ● Pt1000 a=385 -40 to 160C
- Pt100 a=385 0 to 100C ● Pt100 a=392 0 to 100C ● NiFe604 a=518 -80 to 100C
- Pt100 a=385 0 to 200C ● Pt100 a=392 0 to 200C ● NiFe604 a=518 0 to 100C
- Pt100 a=385 0 to 400C ● Pt100 a=392 0 to 400C
- Pt100 a=385 -200 to 200C ● Pt100 a=392 -200 to 200C

Note: IEC RTD 100 ohms.=0.00385
JIS RTD 100 ohms.=0.00392

Thermocouple ranges are:

- B 500 to 1800C ● K 0 to 1370C ● T -100 to 400C
- E 0 to 1000C ● R 500 to 1750C
- J 0 to 760C ● S 500 to 1750C

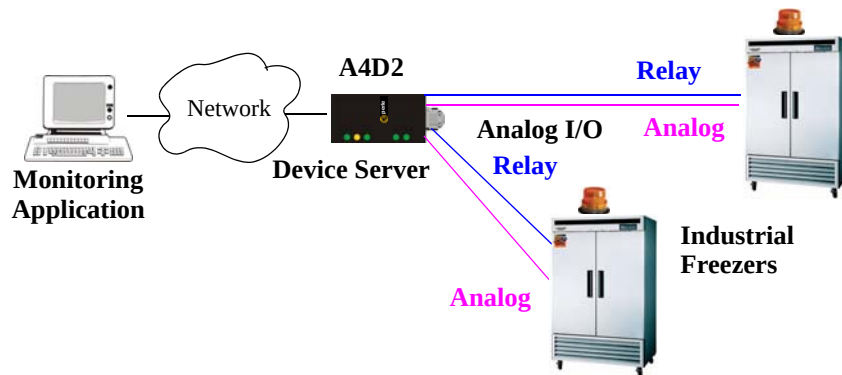
The the following example, a Temperature I/O Device Server is used to monitor industrial freezer as a temperature sensor, with an alarm set to send a syslog message if the temperature rises above 31° C.



Analog Channels

Analog input channels monitor current or voltage within a specified range and can then trigger an alarm for up to five severity levels that will send an email, SNMP trap, and/or syslog message when the alarm is triggered and/or cleared. See [Alarms on page 75](#) for more information on alarms.

In our industrial freezer warehouse, a Device Server A4R2 is used to monitor humidity transducers, which are in place to help prevent freezer burn. If the humidity reaches a certain percentage (monitored by an Analog channel) a syslog message is sent to the Monitoring Application. The Monitoring Application then sends a command to the Device Server via the Perle API that causes the Relay channel to activate an internal freezer dehumidifier. The relay is turned off when the Analog channel sends a clear syslog message to the Monitoring Application and the Relay channel is deactivated.



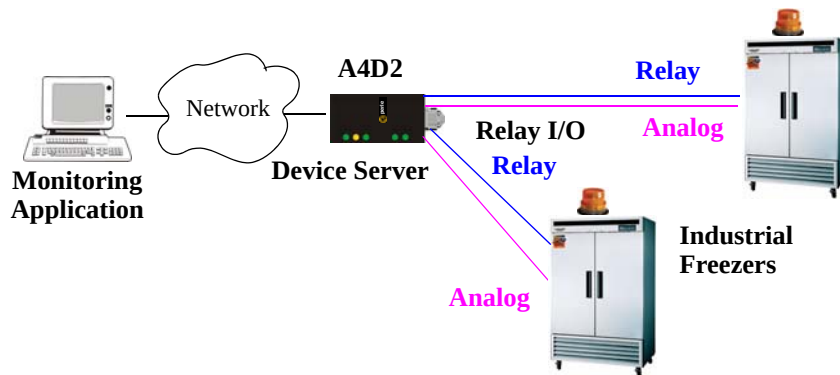
Relay Channels

Relay output channels work as a physical on/off switch, and are used to drive higher voltage devices with a lower controlling voltage.

You can configure the following Relay output channel options:

- You can choose to manually activate/deactivate the Relay output.
- You can choose to manually activate/deactivate the Relay output and then specify that the Relay output will either pulse (you get to specify the active and inactive pulse times) continuously or for a specified number of pulse counts.
- You can choose to manually activate/deactivate the Relay output and then specify a delay before the output goes from inactive to active or active to inactive.
- You can also specify a failsafe action that can either active or inactivate the Relay output when the failsafe timer is triggered (see [Failsafe Timer](#) on page 74 for more information).

In our industrial freezer warehouse, a Device Server A4R2 is used to monitor humidity transducers, which are used to help prevent freezer burn. If the humidity reaches a certain percentage (monitored by an Analog channel) a syslog message is sent to the Monitoring Application, causing the Relay channel to activate an internal freezer dehumidifier. The Relay channel is deactivated when the Analog channel sends a clear syslog message to the Monitoring Application and the Relay channel is deactivated.



Serial Signals

When the **Line Service** is set to **Signal I/O**, you will be able to configure the serial signal I/O pins. This allows you to enable/disable each of the serial pins in the serial port that can be used as Digital inputs and outputs. The serial signal Digital I/O channels can be set for Digital output (RTS and DTR) and/or Digital input (DSR, DCD, and CTS).

SNMP Traps

When you enable SNMP traps for Digital and Analog inputs, a value is returned when an alarm triggers or clears. This section decodes the SNMP trap values.

Value	Alarm	Description
0	IO_DI_ALARM_SENSOR	Trap for the Digital input Trigger.
1	IO_DI_ALARM_SERIAL_DSR	Trap for the Digital input DSR serial pin Trigger.
2	IO_DI_ALARM_SERIAL_DCD	Trap for the Digital input DCD serial pin Trigger.
3	IO_DI_ALARM_SERIAL_CTS	Trap for the Digital input CTS serial pin Trigger.
4	IO_AI_ALARM_LEVEL1	Trap for Analog input Alarm Level 1.
5	IO_AI_ALARM_LEVEL2	Trap for Analog input Alarm Level 2.
6	IO_AI_ALARM_LEVEL3	Trap for Analog input Alarm Level 3.
7	IO_AI_ALARM_LEVEL4	Trap for Analog input Alarm Level 4.
8	IO_AI_ALARM_LEVEL5	Trap for Analog input Alarm Level 5.
9	IO_DI_ALARM_SENSOR_CLEAR	Trap for Digital input trigger Clear Mode.
10	IO_DI_ALARM_SERIAL_DSR_CLEAR	Trap for Digital input DSR serial pin trigger Clear Mode.
11	IO_DI_ALARM_SERIAL_DCD_CLEAR	Trap for Digital input DCD serial pin trigger Clear Mode.
12	IO_DI_ALARM_SERIAL_CTS_CLEAR	Trap for Digital input CTS serial pin trigger Clear Mode.
13	IO_AI_ALARM_LEVEL1_CLEAR	Trap for the Analog input Alarm Level 1 Clear.
14	IO_AI_ALARM_LEVEL2_CLEAR	Trap for the Analog input Alarm Level 2 Clear.
15	IO_AI_ALARM_LEVEL3_CLEAR	Trap for the Analog input Alarm Level 3 Clear.
16	IO_AI_ALARM_LEVEL4_CLEAR	Trap for the Analog input Alarm Level 4 Clear.
17	IO_AI_ALARM_LEVEL5_CLEAR	Trap for the Analog input Alarm Level 5 Clear.

Calibrating Analog Input

To calibrate an Analog input channel, read the section that applies to the type of input you are calibrating. Note that calibration will be done for the active channel configuration; for example, if Channel A1 is set to voltage, you cannot calibrate it for current. The voltage range configured for this channel will also dictate what is being calibrated. For example, if this channel is configured for a range of +/-10V, calibrating this channel will calibrate all channels which are configured for +/-10V. During the calibration process, you will be asked to apply the minimum and maximum configured range value to the channel; for example, to calibrate for voltage +/- 10V, you will be prompted to first apply -10V and then +10V to the channel.

Also, you cannot actively calibrate disabled channels (although, for Voltage, if you enable the channel and then set it for a range that has already been calibrated for another channel, it will also be calibrated).

Calibrating Voltage

When calibrating the Device Server Analog input for voltage, you will need a calibration meter that is better than .1% volts precision. When you calibrate one channel, all voltage channels are automatically calibrated for that range; if another channel is set for a different range, you will need to calibrate that channel separately, but all channels that use that range are also automatically calibrated.

Calibrating Current

When calibrating the Device Server Analog input for current, you will need a calibration meter that is better than .1% current precision. Each channel needs to be calibrated individually.

Calibrating Temperature Input

To calibrate an Analog (Temperature) input channel, read the section that applies to the type of input you are calibrating. Note that calibration will be done for the active channel configuration; for example, if Channel A1 is set to thermocouple, you cannot calibrate it for RTD. During the calibration process, you will be asked to apply the minimum and maximum range value to the channel in either mV or Ohms; for example, to calibrate for thermocouple J 0 to 760C, you will be prompted to first apply -80mV and then +80mV to the channel.

Also, you cannot actively calibrate disabled channels (although if you enable the channel and then set it for the type of thermocouple or RTD that has already been calibrated on another channel, it will also be calibrated).

Calibrating Thermocouple

When calibrating the Device Server Analog input for thermocouple, you will need a calibration meter that is better than .15% accuracy. When you calibrate one channel, all thermocouple channels are automatically calibrated for that range; if another channel is set for a different range, you will need to calibrate that channel separately, but all channels that use that range are automatically calibrated.

Calibrating RTD

When calibrating the Device Server Analog input for RTD, you will need a resistor that is better than .05% Ohms accuracy. When you calibrate one channel, all RTD channels are automatically calibrated for that range; if another channel is set for a different range, you will need to calibrate that channel separately, but all channels that use that range are automatically calibrated.



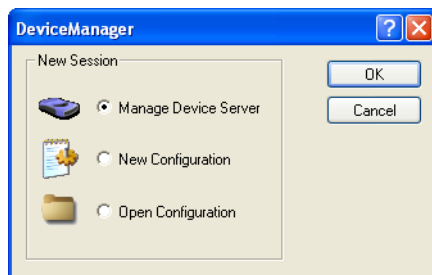
Using the DeviceManager

Introduction

This chapter provides information about configuring/managing the Device Server using the DeviceManager. It is assumed that the DeviceManager has already been installed; if you still need to install the DeviceManager, see [Using DeviceManager on page 39](#).

Starting a New Session

When you start the DeviceManager application, the New Session window is displayed.

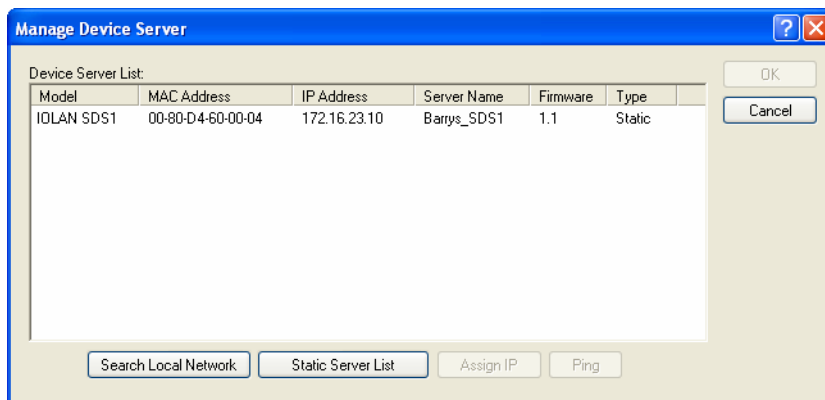


You can choose:

- **Manage Device Server**—Connect to a Device Server to manage/view it.
- **New Configuration**—Create a new Device Server configuration.
- **Open Configuration**—Open an existing configuration file.

Managing a Device Server

You can connect to Device Servers or assign a temporary IP address to a new Device Server. Whenever you connect to a Device Server through the DeviceManager, you connect as the Admin user and must supply the password for the Admin user.



If you want to connect to a Device Server to manage/configure it, or assign a temporary IP address to a Device Server, select **File, New Session** and the **Manage Device Server** radio button. If you want to create a new or edit an existing configuration file, select **File, New Session** and the **New Configuration** or **Open Configuration** radio button.

Populating the Device Server List

The first time you start the DeviceManager, the **Manage Device Server** window will be empty. To add Device Servers to the Device Server **List**, you can do either of the following:

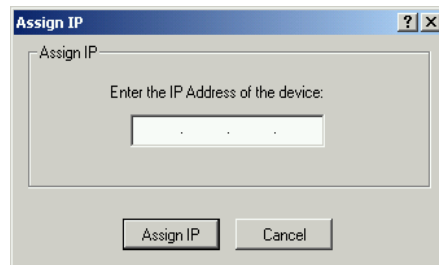
- Click the **Search Local Network** button. This searches the local network segment and automatically displays any Device Servers it finds. Any Device Servers found by this method will be displayed in **Type** column as **Dynamic**. Once you close the DeviceManager, any Device Servers that were displayed as **Dynamic** will not be there until you click the **Search Local Network** button again.
- Click the **Static Server List** button to add Device Servers to the Device Server **List** permanently. This also allows you to add Device Servers that are not found on the local network segment with the **Search Local Network** button. To connect to a Device Server that is not in the Device Server List and resides outside the local network, see [Adding/Deleting Static Device Servers on page 93](#).

For more information about managing a Device Server, see [Managing a Device Server on page 95](#).

Assigning a Temporary IP Address to a New Device Server

If your network does not use DHCP/BOOTP, you can temporarily assign an IP address to a Device Server that is connected to your local network segment, for the purpose of connecting to it and downloading a configuration file (containing a permanent IP address). To temporarily assign an IP address to a Device Server, do the following:

1. Click the **Search Local Network** button. The Device Server will be displayed in the **IP Address** column as **Not Configured**.
2. Select the new Device Server and click the **Assign IP** button. The following window is displayed:

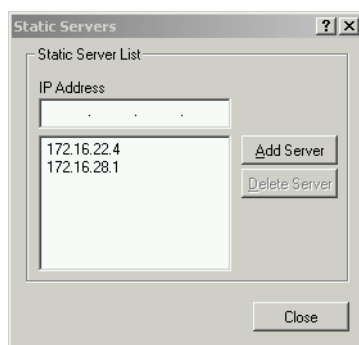


3. Type a valid temporary IP address into the address field and click the **Assign IP** button.
4. Double-click the Device Server in the Device Server **List**. If this is the first time you are accessing the Device Server, type in the factory default Admin password, **superuser**, and click **OK**. The DeviceManager will display a window indicating that it is trying to authenticate and connect you on the Device Server.
5. If the authentication and connection are successful, the Server Info window is displayed. You are now ready to configure the Device Server. If authentication was unsuccessful, try to connect to the Device Server again; you probably mistyped the password for the Admin user.

For more information about managing a Device Server, see [Managing a Device Server on page 95](#).

Adding/Deleting Static Device Servers

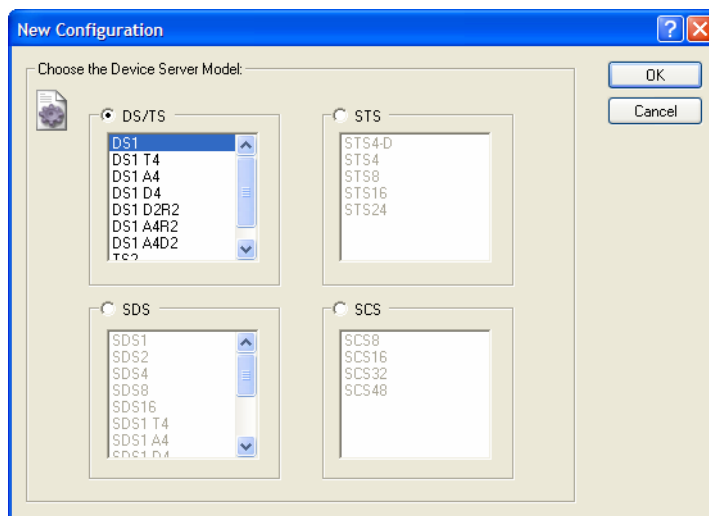
To permanently add or delete a Device Server to/from the Device Server **List**, select the **Static Server List** button. The following window is displayed:



To permanently add a Device Server to the Device Server **List**, type in the IP address of the Device Server and click the **Add Server** button. To permanently delete a Device Server from the Device Server **List**, select the Device Server's IP address and click the **Delete Server** button.

Creating a New Device Server Configuration

If you selected the **New Configuration** radio button, the New Configuration window is displayed.



Select the Device Server model for which you want to create a new configuration file.

Opening an Existing Configuration File

If you selected the **Open Configuration** radio button, a browse window is opened so you can select the configuration file you want to edit. Device Server configuration files can be in the Device Server-native binary format (**.dme**) or as a text file (**.txt**), which can be edited with a text editor. Either configuration version can be imported into the DeviceManager.

Connecting to a Device Server

To connect to a Device Server, double-click on the Device Server in the **Device Server List**. You will be prompted for the Admin Password.



If the authentication and connection are successful, the Device Server's **Server Info** window is displayed.

If you cannot connect to a Device Server, you can highlight the Device Server and click the **Ping** button to verify that the DeviceManager can communicate with the Device Server's IP Address. If the ping times out, then you might need to set up a Gateway in your Device Server or verify that your network is communicating correctly.

Managing a Device Server

Once you are connected to a Device Server, you can edit its configuration, download a new configuration, save the configuration to file, perform administrative tasks, and view statistics about the Device Server and its network environment.

DeviceManager Work Flow

When you connect to a Device Server, the Device Server's configuration is automatically uploaded to the Device Manager. Before you make any changes to the configuration, you probably want to save the configuration locally, to make a backup file of the configuration. Use the navigation panel to select the feature that you want to edit. After you make all your changes to a configuration window, you must click the **Apply** button to submit those changes. When you have completed all of your configuration edits, select **Tools, Download Configuration to Unit**. If you want your changes to take effect at this point, select **Tools, Reboot Server**.

Creating/Editing Configuration Files

You can create and edit Device Server configuration files. When you open a new configuration file, the configuration file contains the Device Server's factory default settings.

Working With the Device Server Configuration

When you connect to a Device Server, the configuration that is saved to FLASH is automatically uploaded to the Device Manager. It is suggested that you save the working configuration to a file as a backup precaution by selecting **Tools, Save Configuration to File**. You can then make any edits to the configuration and download it back to the Device Server by selecting **Tools, Download Configuration to Unit**. The downloaded configuration does not take effect until you reboot the Device Server by selecting **Tools, Reboot Server**. If you want to continue managing/configuring the Device Server, you can reconnect to the Device Server after it has been rebooted.

Working With a Local Configuration File

You can also connect to a Device Server and open a saved configuration file by selecting **Tools, Get Configuration, Import from File**. This configuration can then be edited or just downloaded right to the Device Server by selecting **Tools, Download Configuration to Unit**. The downloaded configuration does not take effect until you reboot the Device Server by selecting **Tools, Reboot Server**. If you want to continue managing/configuring the Device Server, you can reconnect to the Device Server after it has been rebooted.

Configuring the Server

The following sections describe how to configure the Device Server’s server parameters.

Configuring the Main Server Window

When you select **Server Configuration**, **Server** from the navigation panel, the following Server window is displayed.

The screenshot shows a 'Server' configuration window. Under the 'Server' tab, there are input fields for 'Server Name' (set to 'localhost'), 'Domain Name', 'Internet Address [Eth1]' (set to '0 . 0 . 0 . 0'), and 'Mask' (set to '0 . 0 . 0 . 0'). Below these is a 'DHCP/BOOTP' checkbox and an 'Advanced...' button. Under the 'Services' tab, there are checkboxes for 'TelnetD', 'SPCD', 'DeviceManagerD', 'Syslog', 'SNMPD', and 'ModbusD', all of which are checked. 'HTTPD' is also listed but is not checked.

Enter values in the Device Server parameters that you need for your production environment.

Server

- Server Name** You must supply a name for the Device Server.
- Domain Name** Unique name for your domain, your location in the global network. Like Hostname, it is a symbolic, rather than a numerical, identifier.
See [IPv6 Network on page 41](#) for information on how to determine your IPv6 address.
- Internet Address** The Device Server’s unique IPv4 network IP address. If you are using the Device Server in an IPv6 network, this field can be left blank.
- Mask** The network subnet mask. For example, 255.255.0.0.

Services

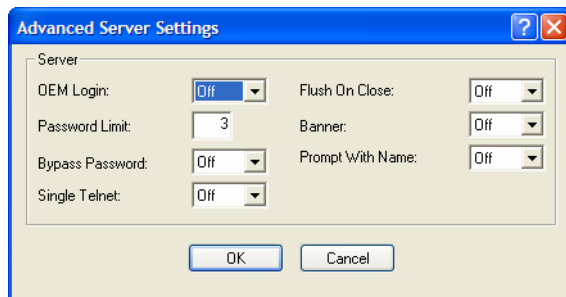
Services are either daemon or client processes that run on the Device Server. You can disable any of the services for security reasons. If you disable the DeviceManagerD service, you will not be able to use DeviceManager to connect to a Device Server.

- TelnetD** Telnet daemon process in the Device Server on port 23.
- SPCD** SPC (Trueport) daemon process in the Device Server on port 688.
- DeviceManagerD** DeviceManager daemon process in the Device Server. If you disable this service, you will not be able to connect to the Device Server with the DeviceManager application. DeviceManagerD listens on port 33812 and sends on port 33813.
- HTTPD** HTTP daemon process in the Device Server on port 80.
- SNMPD** SNMP daemon process in the Device Server on port 161.
- ModbusD** Modbus daemon process in the Device Server on port 502.

Syslog	Syslog client process in the Device Server.
DHCP/BOOTP	DHCP/BOOTP client process in the Device Server. By default, this is disabled/off. If this is enabled, the server IP address parameter is disabled.

Configuring Advanced Server Settings

In the Server window, the following window is displayed when you click the Advanced button.



Configure the appropriate parameters:

OEM Login	When set, and a custom language file is in use, the login prompt will use the string defined in the language file as the login prompt instead of the default prompt, login:.
Password Limit	The number of attempts a user is allowed to enter a password for a port. If this limit is exceeded, the port is disabled for 5 minutes. A user with Admin level rights can restart the port, bypassing the timeout, by issuing a kill on the disabled port. The default value is 3 .
Bypass Password	When set, authorised users who do not have a password set, with the exception of the Admin user, WILL NOT be prompted for a password at login with Local Authentication .
Single Telnet	Sets all reverse connections (raw and telnet) to a one connection at a time mode. Server-side applications will get a (socket) connection refused until: • All data from previous connections on that serial port has drained • There are no other connections • Up to a 1 second interconnection poll timer has expired This also enables a per-connection keepalive TCP keepalive feature. After approximately 3 minutes of network connection idle time, the connection will send a gratuitous ACK to the network peer, thus either ensuring the connection stays active OR causing a dropped connection condition to be recognised by the reverse service (all connections). Applications using Single Telnet need to be aware that there can be some considerable delay between a network disconnection and the port being available for the next connection attempt; this is to allow any data sent on prior connections to be transmitted out of the serial port. Application network retry logic needs to accommodate this feature. The default value is Off.
Flush On Close	When enabled, deletes any pending data when a port is closed; as opposed to maintaining the port to send pending data. The default value is Off .
Banner	This parameter concerns the banner information (product name/software version). This banner information is presented to a user with a login prompt. For security reasons, you can turn off the display of this information. The default is Off .

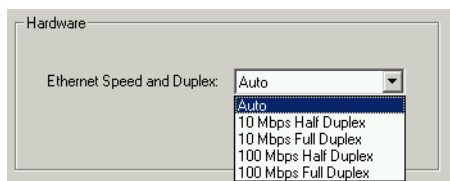
Prompt With Name Displays the **Server Name** field value instead of default product name. When enabled, the **Server Name** is displayed in the Device Server login prompt, CLI prompt, WebManager login screen, and the heading of the Menu. The default value is **Off**.

Configuring TruePort Baud

The TruePort Baud configuration window allows you to map the baud rate coming out of the serial host to another baud rate that will run between the Device Server and the serial device. See [Appendix B, *Utilities* on page 205](#) for more information about TruePort.

Configuring the Hardware

You need to configure the ethernet interface that you are using to connect the Device Server to the LAN.

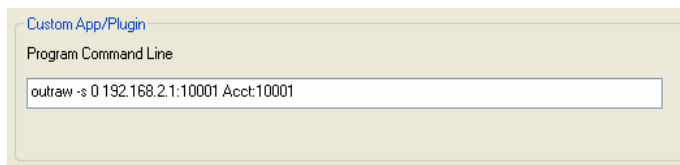


Select the appropriate option:

- Ethernet Speed and Duplex** Define the ethernet connection speed at one of the following:
- **auto**—automatically detects the ethernet interface speed and duplex
 - **10 Mbps Half Duplex**
 - **10 Mbps Full Duplex**
 - **100 Mbps Half Duplex**
 - **100 Mbps Full Duplex**

Custom App/Plugin

You can create a custom application or use a plugin that can run on the Device Server using the Perle SDK.



Configure the following parameter:

Program Command Line The name of the SDK program executable that has been already been downloaded to the Device Server, plus any parameters you want to pass to the program. Maximum of 80 characters. Use the **shell** CLI command as described in the *SDK Programmer's Guide* to manage the files that you have downloaded to the Device Server. For example, using sample outraw program, you would type:

outraw -s 0 192.168.2.1:10001 Acct:10001

if you were starting the application on the Server (notice the **-s 0** parameter specifies Line 1).

Configuring Lines

When you configure the Device Server **Line**, you are specifying how the port will be used and accessed. You can always make changes to **Line** parameters, click the **Apply** button, and then select **Tools, Kill Line** to test your changes. However, you still must select **Tools, Download Configuration to Unit** and **Tools, Reboot Server** to make your changes permanent and take effect.

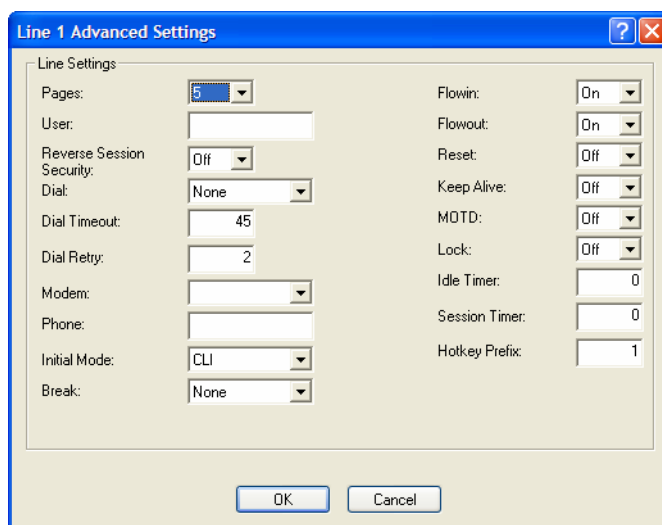
Configure the appropriate parameters:

- | | |
|-------------------------|--|
| Enable Line | Enables/disables the line (available only on the TS2 model). The default is enabled. |
| Line Name | Provide a name for the line so it can be easily identified. |
| Service | Defines the Line Service , which determines how the line will be used.
See Service Settings on page 103 for more information about configuring each Line Service . |
| Internet Address | Used with reverse sessions, users can access serial devices connected to the Device Server by the specified Internet Address (or host name that can be resolved to the Internet Address in a DNS network). You must reboot the Device Server for the Internet Address to take affect (the kill line option does not apply to this parameter). |
| DS Port | The Device Server port number. |

Terminal Type	Specifies the type of terminal connected to the line: • Dumb • WYSE60 • VT100 • ANSI • TVI925 • IBM3151TE • VT320 (specifically supporting VT320-7) • HP700 (specifically supporting HP700/44) • Term1, Term2, Term3 (user defined terminals)
Serial Interface	Specifies the type of line that is being used with the Device Server. Select either EIA-232, EIA-422, or EIA-485 (TS2 supports only EIA-232).
Speed	Specifies the baud rate of the line; keep in mind that speed is affected by the length of the cable. You can also specify a custom baud rate; valid values are 50-230400.
Bits	Specifies the number of bits in a byte. The default is 8.
Parity	Specifies if you are using Even, Odd, or No parity on the line. If you want to force a parity type, you can specify Mark for 1 or Space for 0.
Stop Bits	Specifies the number of stop bits that follow a byte.
Flow Control	Defines whether the data flow is handled by the software (Soft), hardware (Hard), Both, or None.
Single Character Interrupt	When enabled, causes the Device Server to process every character as it comes in, as opposed to buffering the characters before processing; this provides better latency at the expense of efficiency.
Duplex	Specify whether the line is Full Duplex (communication both ways at the same time) or Half Duplex (communication in one direction at a time).
TX Driver Control	Used with a EIA-485 serial interface, if your application supports RTS (Request To Send), select this option. Otherwise, select Auto. Default is Auto.
Echo Suppression	This parameter applies only to EIA-485 Half Duplex mode. All characters will be echoed to the user and transmitted across the serial ports. Some EIA-485 applications require local echo to be enabled in order to monitor the loopback data to determine that line contention has occurred. If your application cannot handle loopback data, echo suppression should be On. The default is echo suppression Off.
Monitor DSR	Specifies whether the RS-232 signal DSR (data set ready) should be monitored. This is used with modems or any device that sends a DSR signal. When it is monitored and the Device Server detects a DSR signal, the line service is started. Default is Off. If both Monitor DCD and Monitor DSR are enabled, both signals must be detected before the line service is started.
Monitor DCD	Specifies whether the RS-232 signal DCD (Data Carrier Detect) should be monitored. This is used with modems or any other device that sends a DCD signal. When it is monitored and the Device Server detects a DCD signal, the line service is started. Default is Off. If both Monitor DCD and Monitor DSR are enabled, both signals must be detected before the line service is started.

Advanced Line Settings

You can configure these advanced settings for a line.



The image shows a dialog box titled "Line 1 Advanced Settings". It contains two columns of settings. The left column includes: Pages (5), User (empty text box), Reverse Session Security (Off), Dial (None), Dial Timeout (45), Dial Retry (2), Modem (empty dropdown), Phone (empty text box), Initial Mode (CLI), and Break (None). The right column includes: Flowin (On), Flowout (On), Reset (Off), Keep Alive (Off), MOTD (Off), Lock (Off), Idle Timer (0), Session Timer (0), and Hotkey Prefix (1). At the bottom are OK and Cancel buttons.

Configure the appropriate parameters:

- | | |
|---------------------------------|--|
| Pages | For DSLogin line service, this is the number of video pages the terminal supports. Valid values are 1-7. The default is 5 pages. |
| User | For DSLogin line service, makes this a line that is dedicated to the specified user. Only this user will be able to log in on this line and they won't need to enter their login name - just their password. |
| Reverse Session Security | Enables/disables login/password authentication, locally or externally, on reverse Telnet connections. The default is Off . |
| Dial | Determines how a modem will work on the line. If your user is remote and will be dialing in via modem or ISDN TA, set this parameter to In ; if the Device Server is being used as a router, set this parameter to either In , Out , or Both , depending on which end of the link your Device Server is situated and how you want to initiate the communication. |
| Dial Timeout | The number of seconds the Device Server will wait to establish a connection to a remote modem. The default value is 45 seconds. |
| Dial Retry | The number of times the Device Server will attempt to establish a connection with a remote modem. The default value is 2 . |
| Modem | The name of the predefined modem that is used on this line. |
| Phone | The phone number to use when Dial is set to Out . |
| Initial Mode | Specifies the initial interface a user navigates when logging into the line; either the Menu or a prompt for the CLI . The default is CLI . |

Break	Specifies how a break is interpreted: • None—The Device Server ignores the break key completely and it is not passed through to the host. This is the default setting. • Local—The Device Server deals with the break locally. If the user is in a session, the break key has the same effect as a hot key. • Remote—When the break key is pressed, the Device Server translates this into a telnet break signal which it sends to the host machine. • Brkintr—On some systems such as SunOS, XENIX, and AIX, a break received from the peripheral is not passed to the client properly. If the client wishes to make the break act like an interrupt key (for example, when the stty options -ignbrk and brkintr are set).
Flowin	Determines if input flow control is to be used. Default is On . This is active only when Line Flow Control is set to Soft , Hard , or Both .
Flowout	Determines if output flow control is to be used. Default is On . This is active only when Line Flow Control is set to Soft , Hard , or Both .
Reset	Resets the terminal type connected to the line when a user logs out.
Keep Alive	Enables a per-connection TCP keepalive feature; after approximately 3 minutes of network connection idle time, the connection will send a gratuitous ACK to the network peer, either ensuring the connection stays active OR causing a dropped connection condition to be recognised by the reverse raw service. Applications using this feature need to be aware that there might be some considerable delay between a network disconnection and the port being available for the next connection attempt; this is to allow any data sent on prior connections to be transmitted out of the serial port buffer. Application network retry logic needs to accommodate this feature.
MOTD	Enables/disables the message of the day on the line.
Lock	When enabled, the user can lock his terminal with a password using the Hotkey Prefix (default Ctrl-a) ^a l (lowercase L). The Device Server prompts the user for a password and a confirmation.
Idle Timer	Enter a time period, in seconds, for which the Idle Timer will run. Use this timer to close a connection because of inactivity. When the Idle Timer expires, the Device Server will end the connection. The maximum value is 4294967 seconds (about 49 days). The default value of 0 (zero) means the Idle Timer will not expire, so the connection is permanently open.
Session Timer	Enter a time, in seconds, for which the Session Timer will run. Use this timer to forcibly close the session (connection). When the Session Timer expires, the Device Server will end the connection. The default value is 0 seconds so the port will never timeout. The maximum value is 4294967 seconds (about 49 days).

Hotkey Prefix

The prefix that a user types to lock a line or redraw the Menu. The default value is **hex 01**, which corresponds to **Ctrl-a (^a)** (hex value 02 would be Ctrl-b (^b), etc.):

- **^a l**—(Lowercase L) Locks the line until the user unlocks it. The user is prompted for a password (any password, excluding spaces) and locks the line. Next, the user must retype the password to unlock the line.
- **^r**—When you switch from a session back to the Menu, the screen may not be redrawn correctly. If this happens, use this command to redraw it properly.

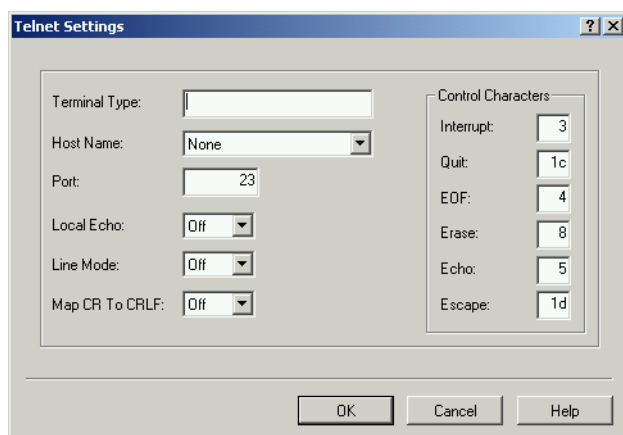
You can use the **Hotkey Prefix** key to lock a line only when the **Line Lock** parameter is **On**.

Service Settings

Line Services determine how line is defined. As a rule, when you are accessing a serial device through the Device Server, coming from the ethernet side, you want to set the **Line Service** to **Reverse Raw** or **Reverse Telnet**.

DSLogin

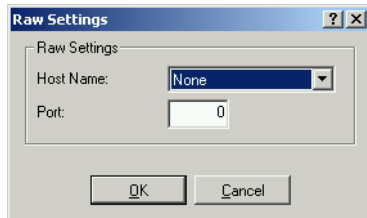
When the **Line Service** is set to **DSLogin**, any user accessing the Device Server will have to log into the Device Server. What happens after the user successfully logs into the Device Server is based on how the user is configured. For example, if after a successful login, the user is set to telnet to a specific host, you will want to set the Telnet parameters that will be used by the user for the telnet session (any parameters that are also available in the user's configuration are overridden by the user's definitions).



See [Telnet Settings](#) on page 104 for information on the Telnet Settings parameters.

Raw Settings

When the **Line Service** is set to **Direct** or **Silent Raw**, data is sent through the connection in its original format. This raw TCP/IP connection is initiated from the Device Server to the configured host.

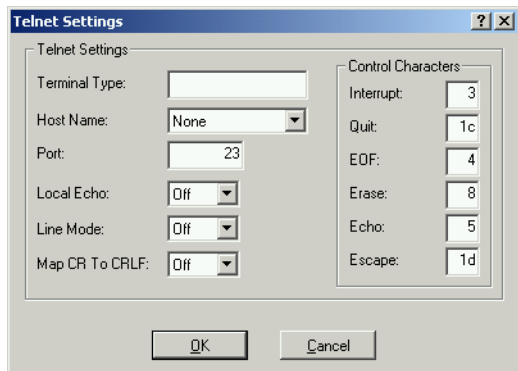


Configure the following parameters:

- Host Name** The name of the target host.
- Port** The port number the target host is listening on for incoming connections.

Telnet Settings

When the **Line Service** is set to **Direct** or **Silent Telnet**, data is sent through the connection in a telnet session. This telnet session is initiated from the Device Server to the configured host.



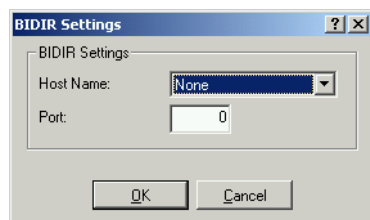
Configure the following parameters:

- Terminal Type** Type of terminal attached to this line; for example, ANSI or WYSE60.
- Host Name** The name of the target host.
- Port** The port number the target host is listening on for incoming connections.
- Local Echo** Toggles between local echo of entered characters and suppressing local echo. Local echo is used for normal processing, while suppressing the echo is convenient for entering text that should not be displayed on the screen, such as passwords. This parameter can only be used when **Line Mode** is **On**. Default is **Off**.
- Line Mode** When **On**, keyboard input is not sent to the remote host until **Enter** is pressed, otherwise input is sent every time a key is pressed. Default is **Off**.
- Map CR to CRLF** Maps carriage returns (CR) to carriage return line feed (CRLF). The default value is **Off**.

Interrupt	Defines the interrupt character. Typing the interrupt character interrupts the current process. This value is in hexadecimal with a default value of 3 (ASCII value ^C).
Quit	Defines the quit character. Typing the quit character closes and exits the current telnet session. This value is in hexadecimal with a default value of 1c (ASCII value FS).
EOF	Defines the end-of-file character. When Line Mode is On, entering the eof character as the first character on a line sends the character to the remote host. This value is in hexadecimal with a default value of 4 (ASCII value ^D).
Erase	Defines the erase character. When Line Mode is Off, typing the erase character erases one character. This value is in hexadecimal with a default value of 8 (ASCII value ^H).
Echo	Defines the echo character. When Line Mode is On, typing the echo character echoes the text locally and sends only completed lines to the host. This value is in hexadecimal with a default value of 5 (ASCII value ^E).
Escape	Defines the escape character. Returns you to the command line mode. This value is in hexadecimal with a default value of 1d (ASCII value GS).

BIDIR Settings

When the **Line Service** is set to **BIDIR**, a bidirectional connection is created, with data flowing in both directions in its original format. This raw TCP/IP connection can be initiated from either the Device Server or the configured host.



Configure the following parameters:

Host Name	The name of the target host.
Port	The port number the target host is listening on for incoming connections.

UDP Settings

When the **Line Service** is set to **UDP**, the Device Server processes UDP packets according to the UDP settings.

Configure the following parameters:

- | | |
|-------------------------|--|
| Start IP Address | The first host IP address in the range of IP addresses (for IPV4 or IPV6) that the Device Server will listen for messages from and/or send messages to. |
| End IP Address | The last host IP address in the range of IP addresses (for IPV4, not required for IPV6) that the Device Server will listen for messages from and/or send messages to. |
| Port | The port that the Device Server will use to relay messages to servers/hosts. The Device Server will listen for UDP packets on the port configured by the DS Port parameter. |
| Direction | <p>The direction in which information is received or relayed:</p> <ul style="list-style-type: none"> ● None—UDP service not enabled. ● In—LAN to serial. ● Out—Serial to LAN. ● Both—Messages are relayed both directions. |

VModem Settings

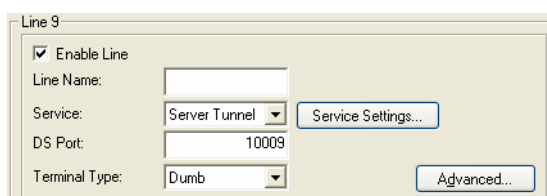
When the **Line Service** is set to **VModem**, the Device Server acts as a virtual modem. After a virtual modem connection is established, data will flow in both directions in its original format.

Configure the following parameters:

Host Name	The target host name.
Port	The port number the target host is listening on for messages.
Success	String that is sent to the serial device when a connection succeeds. If no string is entered, then the string CONNECT will be sent with the connecting speed, for example CONNECT 9600 .
Failure	String that is sent to the serial device when a connection fails. If no string is entered, then the string NO CARRIER will be sent.
Suppress	If set to No , connection success/failure indication strings are sent to the connected device, otherwise these indications are suppressed.
Response Delay	The amount of time, in milliseconds, before an AT response is sent to the requesting device. The default is 250 ms.
Style	One of the following: • Verbose—Return codes (strings) are sent to the connected device. • Numeric—The following characters can be sent to the connected device: 1 Successfully Connected 2 Failed to Connect 4 Error

Server Tunnel Settings

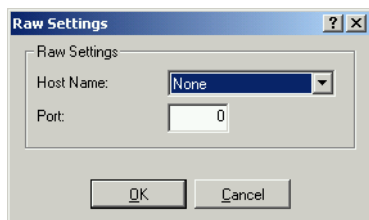
The purpose of the serial **Line Service Client/Server Tunnel** is to allow two Device Servers that are connected back-to-back over Ethernet to virtually link two serial ports. The serial device that initiates the connection is the **Client Tunnel** and the recipient is the **Server Tunnel**, although once the communication tunnel has been successfully established, the communication tunnel will stay connected and can go both ways. The Server Tunnel will support Telnet Com Port Control protocol as detailed in RFC 2217. See [Serial Tunnel Settings on page 66](#) for more information about how to configure the Device Server for a serial tunnelling.



It is important that the **Client Tunnel Port** parameter reflect the **DS Port** set for the Line when the Device Servers are being used back-to-back over Ethernet.

Client Tunnel Settings

The purpose of the serial **Line Service Client/Server Tunnel** is to allow two Device Servers that are connected back-to-back over Ethernet to virtually link two serial ports. The serial device that initiates the connection is the **Client Tunnel** and the recipient is the **Server Tunnel**, although once the communication tunnel has been successfully established, the communication tunnel will stay connected and can go both ways. See [Serial Tunnel Settings on page 66](#) for more information about how to configure the Device Server for a serial tunnelling.

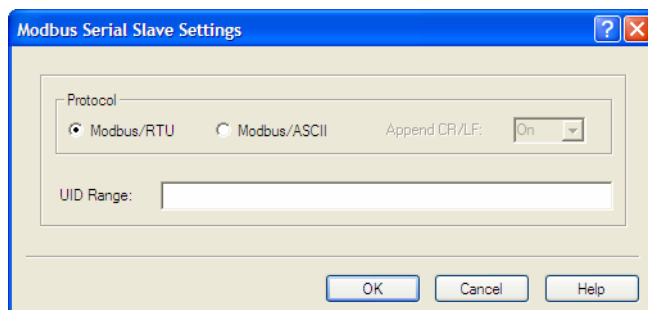


Configure the following parameters:

- | | |
|------------------|--|
| Host Name | The name of the Device Server that is connected to the serial device, acting as the Server Tunnel. |
| Port | The DS Port of the Device Server that is connected to the serial device. |

Modbus Slave Settings

This window configures the parameters for Modbus Slaves residing on the serial side of the Device Server. See [Modbus Configuration on page 57](#) for more information about how to configure the Device Server for a Modbus environment.

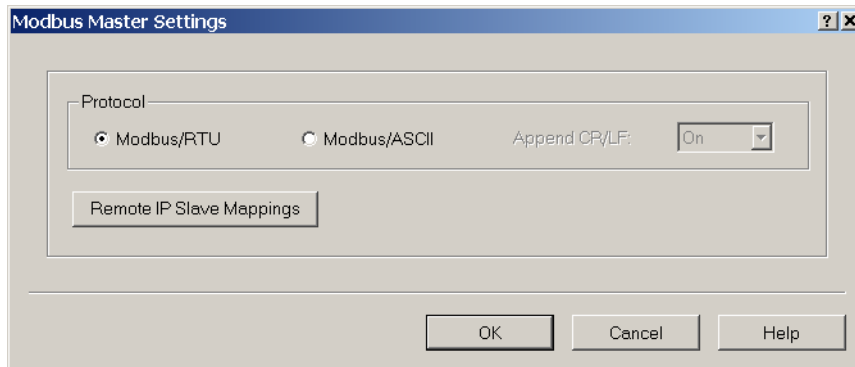


Configure the following parameters:

- | | |
|---------------------|--|
| Modbus/RTU | Select this option if the Modbus Master is configured using the Modbus/RTU protocol. |
| Modbus/ASCII | Select this option if the Modbus Master is configured using the Modbus/ASCII protocol. |
| Append CR/LF | When Modbus/ASCII is selected, adds a CR/LF to the end of the transmission; most Modbus devices require this option. The default is On . |
| UID Range | You can specify a range of UIDs (1-247), in addition to individual UIDs. The format is comma delimited; for example, 2-35, 50, 100-103. |

Modbus Master Settings

This window configures the parameters for Modbus Masters on the serial side of the Device Server. You can also choose to transmit the Modbus Master data encrypted via SSL/TLS. See [Modbus Configuration on page 57](#) for more information about how to configure the Device Server for a Modbus environment.



Configure the following parameters:

- | | |
|--|---|
| Modbus/RTU | Select this option if the Modbus Slave is configured using the Modbus/RTU protocol. |
| Modbus/ASCII | Select this option if the Modbus Slave is configured using the Modbus/ASCII protocol. |
| Append CR/LF | When Modbus/ASCII is selected, adds a CR/LF to the end of the transmission; most Modbus devices require this option. The default is On . |
| Remote IP Slave Mappings Button | Click this button to launch the Remote Slave IP Settings window, where you can configure the TCP/Ethernet Modbus Slaves that the Modbus Master on the Line will communicate with. |

Remote IP Slave Mappings

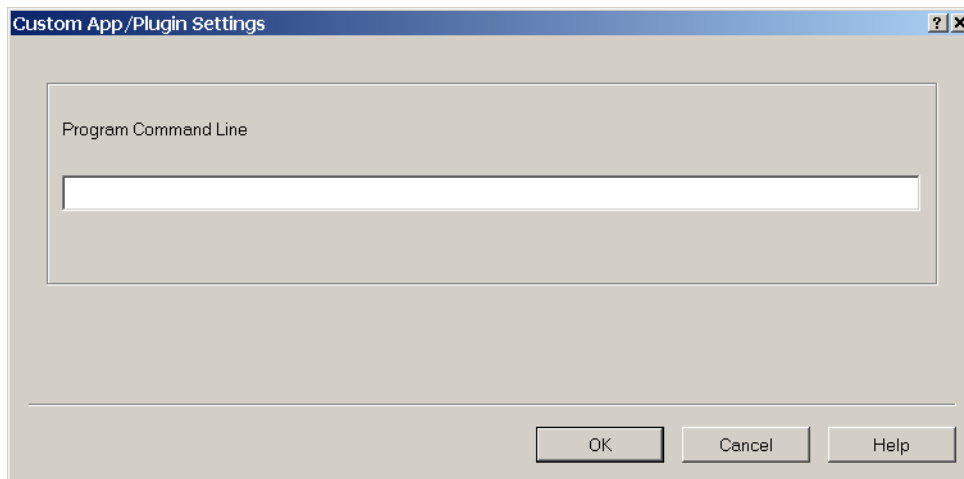
This window allows you to configure all the Modbus Slaves, which reside on the Ethernet/TCP side of the Device Server, that will be receiving messages from the Modbus Master. See [Modbus Configuration on page 57](#) for more information about how to configure the Device Server for a Modbus environment.

Configure the following parameters:

- | | |
|------------------------|--|
| Remote Slave IP | The IP address of the TCP/Ethernet Modbus Slave. |
| Protocol | Specify the protocol that is used between the Modbus Master and Modbus Slave(s), either TCP or UDP. |
| Port | The destination port of the remote Modbus TCP Slave that the Device Server will connect to. |
| Start UID | When Range Mode is Host and you have sequential Modbus Slave IP addresses (for example, 10.10.10.1, 10.10.10.2, 10.10.10.3, etc.), you can specify a UID range and the Device Server will automatically increment the last digit of the configured IP address. Therefore, you can specify a UID range of 1-100, and the Device Server will route Master Modbus messages to all Modbus Slaves with IP addresses of 10.10.10.1 - 10.10.10.100. |
| End UID | When Range Mode is Host and you have sequential Modbus Slave IP addresses (for example, 10.10.10.1, 10.10.10.2, 10.10.10.3, etc.), you can specify a UID range and the Device Server will automatically increment the last digit of the configured IP address. Therefore, you can specify a UID range of 1-100, and the Device Server will route Master Modbus messages to all Modbus Slaves with IP addresses of 10.10.10.1 - 10.10.10.100. |
| Range Mode | If you specify Host , the IP address is used for the first UID specified in the range. The last octet in the IPv4 address is then incremented for subsequent UID's in that range. The Host option is not applicable for IPv6 addresses. If you specify Gateway , the Modbus Master Gateway will use the same IP address when connecting to all the remote Modbus slaves in the specified UID range. |

Custom App/Plugin Settings

You can create a custom application or use a plugin that can run on a specific serial line in Device Server using the Perle SDK.



Configure the following parameter:

**Program
Command Line**

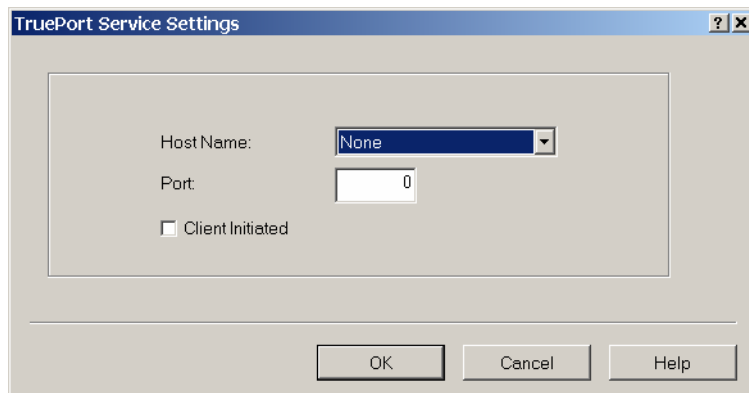
The name of the SDK program executable that has been already been downloaded to the Device Server, plus any parameters you want to pass to the program. Maximum of 80 characters. Use the **shell** CLI command as described in the *SDK Programmer's Guide* to manage the files that you have downloaded to the Device Server. For example, using sample outraw program, you would type:

outraw 192.168.2.1:10001 Acct:10001

if you were starting the application on a line.

TruePort Settings

When the **Line Service** is set to **TruePort**, data is sent through the connection in its original format. This raw TCP/IP connection can be initiated from the Device Server to the configured host or from the host to the Device Server, depending on the settings.



Configure the following parameters:

- | | |
|-------------------------|--|
| Host Name | The name of the target host. |
| Port | The port number the target host is listening on for incoming connections. |
| Client Initiated | When enabled, allows the TruePort client to initiate communication to the Device Server. |

Packet Forwarding

The Packet Forwarding feature allows you to control how the data coming from a serial device is packetized before forwarding the packet onto the LAN network.

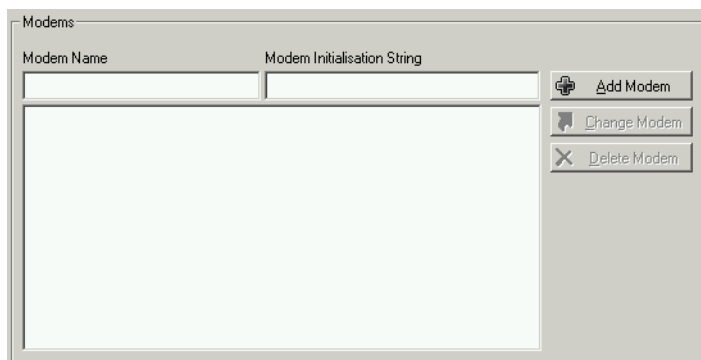
Configure the following parameters:

- | | |
|-------------------------------|---|
| Packet Definition | This section allows you to set a variety of packet definition options. The first criteria that is met causes the packet to be transmitted. For example, if you set a Force Transmit Timer of 1000 ms and a Packet Size of 100 bytes, whichever criteria is met first is what will cause the packet to be transmitted. |
| Packet Size | The number of byte that must be received from the serial port before the packet is transmitted to the network. A value of zero (0) ignores this parameter. Valid values are 0-1024 bytes. The default is 0. |
| Idle Time | The amount of time, in milliseconds, that must elapse between characters before the packet is transmitted to the network. A value of zero (0) ignores this parameter. Valid values are 0-65535 ms. The default is 0. |
| Force Transmit Timer | When the specified amount of time, in milliseconds, elapses after the first character is received from the serial port sender, the packet is transmitted. A value of zero (0) ignores this parameter. Valid values are 0-65535 ms. The default is 0. |
| End Trigger1 Character | When enabled, specifies the character that when received will define when the packet is ready for transmission. The actual transmission of the packet is based on the Trigger Forwarding Rule. Valid values are in hex 0-FF. The default is 0. |
| End Trigger2 Character | When enabled, creates a sequence of characters that must be received to specify when the packet is ready for transmission (if the End Trigger1 character is not immediately followed by the End Trigger2 character, the Device Server waits for another End Trigger1 character to start the End Trigger1/End Trigger2 character sequence). The actual transmission of the packet is based on the Trigger Forwarding Rule. Valid values are in hex 0-FF. The default is 0. |

Frame Definition	This section allows you to control the frame that is transmitted by defining the start and end of frame character(s). If the internal buffer (1024 bytes) is full before the EOF character(s) are received, the packet will be transmitted and the EOF character(s) search will continue. The default frame definition is SOF=00 and EOF=00.
SOF1 Character	When enabled, the Start of Frame character defines the first character of the frame, any character(s) received before the Start of Frame character is ignored. Valid values are in hex 0-FF. The default is 0.
SOF2 Character	When enabled, creates a sequence of characters that must be received to create the start of the frame (if the SOF1 character is not immediately followed by the SOF2 character, the Device Server waits for another SOF1 character to start the SOF1/SOF2 character sequence). Valid values are in hex 0-FF. The default is 0.
Transmit SOF Character(s)	When enabled, the SOF1 or SOF1/SOF2 characters will be transmitted with the frame. If not enabled, the SOF1 or SOF1/SOF2 characters will be stripped from the transmission.
EOF1 Character	Specifies the End of Frame character, which defines when the frame is ready to be transmitted. The actual transmission of the frame is based on the Trigger Forwarding Rule. Valid values are in hex 0-FF. The default is 0.
EOF2 Character	When enabled, creates a sequence of characters that must be received to define the end of the frame (if the EOF1 character is not immediately followed by the EOF2 character, the Device Server waits for another EOF1 character to start the EOF1/EOF2 character sequence), which defines when the frame is ready to be transmitted. The actual transmission of the frame is based on the Trigger Forwarding Rule. Valid values are in hex 0-FF. The default is 0.
Trigger Forwarding Rule	Determines what is included in the Frame (based on the EOF1 or EOF1/EOF2) or Packet (based on Trigger1 or Trigger1/Trigger2). Choose one of the following options: ● Strip-Trigger—Strips out the EOF1, EOF1/EOF2, Trigger1, or Trigger1/Trigger2, depending on your settings. ● Trigger—Includes the EOF1, EOF1/EOF2, Trigger1, or Trigger1/Trigger2, depending on your settings. ● Trigger+1—Includes the EOF1, EOF1/EOF2, Trigger1, or Trigger1/Trigger2, depending on your settings, plus the first byte that follows the trigger. ● Trigger+2—Includes the EOF1, EOF1/EOF2, Trigger1, or Trigger1/Trigger2, depending on your settings, plus the next two bytes received after the trigger.

Configuring Modems

You need to configure a modem if there is a modem connected to the Device Server.

A screenshot of a software window titled "Modems". The window has a light gray border and a white background. At the top, there are two labels: "Modem Name" and "Modem Initialisation String". Below these labels are two text input fields. To the right of the input fields, there are three buttons stacked vertically: "Add Modem" (with a plus icon), "Change Modem" (with a right-pointing arrow icon), and "Delete Modem" (with an X icon). Below the input fields and buttons is a large, empty rectangular area, likely a list or table for displaying configured modems.

Configure the following parameters:

Modem Name The name of the modem. Do not use spaces.

Modem Initialisation String The initialisation string of the modem; see your modem's documentation.

Configuring I/O

This configuration entry will appear in the DeviceManager only when connected to a Device Server that supports I/O or when an I/O model configuration is selected.

Global Settings

The I/O Global Settings enable/disable options that support all I/O channels on your Device Server.

I/O Global Settings

Temperature Scale: Celsius

Failsafe Timer

Enable I/O Failsafe Timer

Timeout: 30 Seconds

Modbus

Enable I/O Modbus Slave

UID: 255

UDP

Enable UDP Broadcast of I/O Status

Broadcast Interval: 30 Seconds

Settings...

TruePort

Enable I/O TruePort Services

Listening Port: 33816 TCP

TruePort I/O Services

I/O Access via API: Disabled

Modbus Serial I/O Access: Disabled

Temperature Settings

Temperature Scale

Select the temperature scale that will be used to display temperature data, either Fahrenheit or Celsius. The default is Celsius.

This option is for Temperature channels only.

Failsafe Timer Settings

The **Failsafe Timer** is activated when no I/O operation occurs within the specified amount of time; I/O operations include everything that can be done manually in the **I/O Status/Control** or programmatically.

Enable I/O Failsafe Timer

Enables/disables the **Failsafe Timer**. This is the global setting that must be enabled to set the **Failsafe Action** on the channel for digital outputs and relays. When this timer expires because of no I/O activity within the specified time interval, the **Failsafe Action** set for the channel determines the action on the output.

Timeout

The number of seconds that must elapse with no I/O activity before the channel **Failsafe Action** is triggered. Valid values are 1-9999. The default is 30 seconds.

117

Modbus Settings

Enabling the Modbus option makes the Device Server act as a Modbus Slave, allowing Modbus Masters to communicate with the Device Server to control and/or retrieve I/O data.

Enable I/O Modbus Slave Enables/disables the ability to control/monitor I/O channels using the Modbus protocol.

UID This is the UID you are assigning to the Device Server, which is acting as a Modbus slave.

TruePort Settings

These TruePort settings pertain specifically to using TruePort to allow serial Modbus Masters to access Device Server I/O data over the network or allowing a serial application to access the Device Server I/O data over the network using the Perle API (see [API I/O Access Over TruePort](#) on page 206 for more information).

Enable I/O TruePort Service Enables/disables serial Modbus application access to the I/O over the network using the TruePort COM redirector feature.

Listening Port The TCP port that the Device Server will listen to for I/O channel data requests from TruePort.

I/O Access via API Displays the access status of being able to access the I/O data via a custom application using the Perle API.

Modbus Serial I/O Access Displays the status of a serial Modbus Master being able to access the Device Server over the network to access I/O data.

UDP Settings

The I/O UDP broadcast feature periodically broadcasts the state of the I/O status in a UDP message.

Enable UDP Broadcast of I/O Status Enables/disables UDP broadcast of I/O channel status (data).

Broadcast Interval Enter the interval, in seconds, for UDP broadcasts of I/O channel status (data). Valid values are 1-9999. Default value is 30 seconds.

Setting Button Click the **Settings** button to access the I/O UDP Broadcast window, where you can define the IP addresses of the receivers of the UDP broadcast.

The I/O UDP settings window allows you to configure the UDP broadcast recipients.

The screenshot shows the 'I/O UDP Settings' dialog box. It features a blue title bar with the text 'I/O UDP Settings' and standard window controls. The main area is divided into four sections, each for a UDP entry. Each section contains a checkbox, two text boxes for IP addresses ('Start IP Address' and 'End IP Address'), and a text box for the 'Port'. All IP addresses are set to '0.0.0.0' and the port is set to '0'. The first entry is selected with a checked checkbox, while the others are unchecked. At the bottom of the dialog are 'OK' and 'Cancel' buttons.

Configure the following parameters:

- | | |
|-------------------------|---|
| UDP Entry | When enabled, broadcasts I/O status (data) to the specified range of IP addresses. |
| Start IP Address | The first host IP address in the range of IP addresses (for IPV4 or IPV6) that the Device Server will listen for messages from and/or send messages to. |
| End IP Address | The last host IP address in the range of IP addresses (for IPV4, not required for IPV6) that the Device Server will listen for messages from and/or send messages to. |
| Port | The UDP port that the Device Server will use to relay messages to servers/hosts. |

Channels

Digital Output

When the channel is set for Digital output, either voltage is applied to the channel or the channel is grounded. Note that the internal jumpers must match the software setting and must be set to **Output** (by default, they are set to Input); see [Digital I/O Module on page 33](#) to find out how to set the internal jumpers.

Configure the following parameters:

- | | |
|-----------------------|--|
| Enable Channel | Enables the channel, allowing the settings to become active. |
| Description | Provide a description of the channel, making it easier to identify. The channel description can be up to 20 characters. |
| Digital Mode | Specify whether the channel will drive the line (output) or will be reading the status of the line (input). The default is Input . The internal jumpers must match the software configuration, so if you change this setting to Output , you will have to also change the internal hardware jumpers. |
| Type | <p>Specify the type of digital output:</p> <ul style="list-style-type: none"> ● Sink—Specifies that the channel will be grounded when active. ● Source—Specifies that the channel will provide voltage when active. ● Sink and Source—Specifies that channel will be grounded when it is inactive and will provide voltage when it is active. <p>The default is Sink.</p> |
| Output | <p>Specify how the channel Digital output will be handled:</p> <ul style="list-style-type: none"> ● Manual—You must manually activate and deactivate the output. ● Pulse—When the output is activated, this configuration will cause the output to pulse (go active and inactive) at a configured rate. ● Inactive-to-Active Delay—The channel output will remain inactive for the specified time interval after it is manually started. ● Active-to-Inactive Delay—The channel output will go inactive after the specified time interval after it is manually started. <p>The default is Manual.</p> |

Pulse Mode	When the Output is Pulse , you can have it pulse in a Continuous manner or specify a pulse Count (each count consists of an active/inactive sequence). The default is Continuous .
Pulse Count	The channel output will pulse for the specified number of times; each count consists of an active/inactive sequence. The default is 1.
Inactive Signal Width	How long the channel output will remain inactive during pulse mode. Valid values are 1-9999 x 100 ms. The default is 100 ms.
Active Signal Width	How long the channel output will be active during the pulse mode. Valid values are 1-9999 x 100 ms. The default is 100 ms.
Failsafe Action	When there has been no I/O activity within the specified time (set in the Global Settings) and the Failsafe Timer is triggered, you can set the Failsafe Action to: • None—The state of the Digital/Relay output remains the same, no change. • Activate Output—Activates the output. • Deactivate Output—Deactivates the output.

Digital Input

When the channel is set for Digital input, it monitors voltage or current. Note that the internal jumpers must match the software setting and must be set to **Input**, which is the default; see [Digital I/O Module on page 33](#) to find out how to set the internal jumpers.

Configure the following parameters:

Enable Channel	Enables the channel, allowing the settings to become active.
Description	Provide a description of the channel, making it easier to identify. The channel description can be up to 20 characters.
Digital Mode	Specify whether the channel will drive the line (output) or will be reading the status of the line (input). The default is Input . The internal jumpers must match the software configuration, so if you change this setting to Output , you will have to also change the internal hardware jumpers.
Latch	Latches (remembers) the activity transition (active to inactive or inactive to active). The default is None.

Invert Signal	Inverts the actual condition of the I/O signal in the status; therefore, an inactive status will be displayed as active.
Trigger	When the trigger condition is met, triggers the specified alarm action. Triggers can be: ● Disabled—No alarm settings. This is the default. ● Inactive—When the expected Digital input is active, going inactive will trigger an alarm. ● Active—When the expected Digital input is inactive, going active will trigger an alarm.
Clear Mode	Specify Manual to manually clear an alarm. Specify Auto to automatically clear the alarm when the trigger condition changes; for example, if the Trigger is Inactive and the alarm is triggered, once the input becomes active again, the alarm will be cleared when Auto is set. The default is Auto .
Syslog	Sends a message to syslog when an alarm is triggered or cleared. The syslog entry includes the severity level and the value that caused the alarm to trigger or clear. The syslog message is associated with Level Critical .
SNMP	Sends an SNMP trap when an alarm is triggered or cleared. The trap consists of the severity level and whether the alarm was triggered or cleared.

Relay

Relay channels can open or close a contact for a higher voltage circuit using a lower level control voltage.

Configure the following parameters:

- | | |
|------------------------------|--|
| Enable Channel | Enables the channel, allowing the settings to become active. |
| Description | Provide a description of the channel, making it easier to identify. The channel description can be up to 20 characters. |
| Output | <p>Specify how the channel Digital output will be handled:</p> <ul style="list-style-type: none"> • Manual—You must manually activate and deactivate the output. • Pulse—When the output is activated, this configuration will cause the output to pulse (go active and inactive) at a configured rate. • Inactive-to-Active Delay—The channel output will remain inactive for the specified time interval after it is manually started. • Active-to-Inactive Delay—The channel output will go inactive after the specified time interval after it is manually started. <p>The default is Manual.</p> |
| Pulse Mode | When the Output is Pulse , you can have it pulse in a Continuous manner or specify a pulse Count (each count consists of an active/inactive sequence). The default is Continuous . |
| Pulse Count | The channel output will pulse for the specified number of times; each count consists of an active/inactive sequence. The default is 1. |
| Inactive Signal Width | How long the channel output will remain inactive during pulse mode. Valid values are 1-9999 x 100 ms. The default is 100 ms. |
| Active Signal Width | How long the channel output will be active during the pulse mode. Valid values are 1-9999 x 100 ms. The default is 100 ms. |
| Failsafe Action | <p>When there has been no I/O activity within the specified time (set in the Global Settings) and the Failsafe Timer is triggered, you can set the Failsafe Action to:</p> <ul style="list-style-type: none"> • None—The state of the Digital/Relay output remains the same, no change. • Activate Output—Activates the output. • Deactivate Output—Deactivates the output. |

Analog

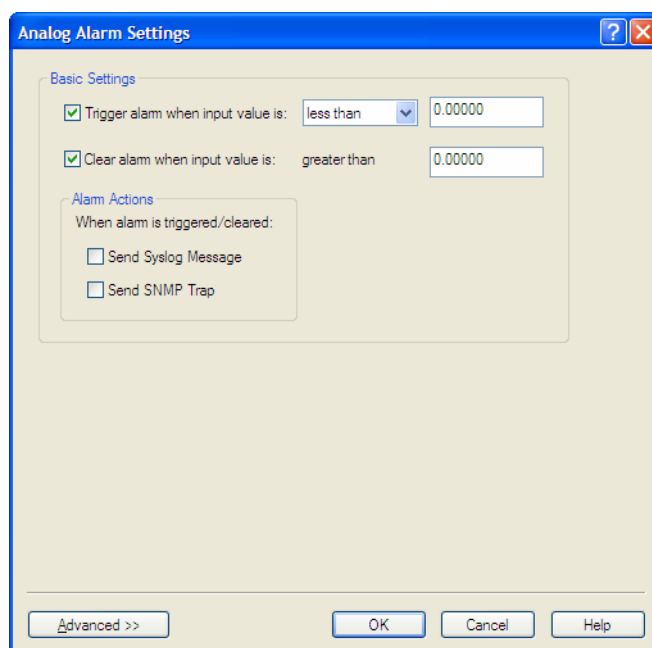
Analog channels monitor current/voltage input. Note that the internal jumpers must match the software setting (by default, they are set to Current); see [Analog Input Module on page 34](#) to find out how to set the internal jumpers.

Configure the following parameters:

- | | |
|-----------------------|---|
| Enable Channel | Enables the channel, allowing the settings to become active. |
| Description | Provide a description of the channel, making it easier to identify. The channel description can be up to 20 characters. |
| Type | Select the type of input being measured, either Current or Voltage . The default is Current . |
| Range | Select the range for the measurement type. The default for Current is 0 - 20mA . The default for Voltage is +/- 10V . |

Basic Alarm Settings

The basic Analog Alarm Settings window allows you to configure one severity alarm level, whereas the advanced window allows you to configure up to five severity alarm levels.



Configure the following parameters:

Trigger alarm when input value is Specify the value that will trigger an alarm, the measurement is based on the **Type** and **Range** that you specify. This value must not fall within the scope of the value used to clear an alarm.

Clear alarm when input value is Specify that value that will clear an alarm, the measurement is based on the **Type** and **Range** that you specify. This value must not fall within the scope of the value used to trigger an alarm.

Send Syslog Alert Sends a message to syslog when an alarm is triggered or cleared. The syslog entry includes the severity level and the value that caused the alarm to trigger or clear. The syslog message is associated with **Level Critical**.

Send SNMP Alert Sends an SNMP trap when an alarm is triggered or cleared. The trap consists of the severity level and whether the alarm was triggered or cleared.

Advanced Alarm Settings

The advanced Analog Alarm Settings window expands the basic alarm settings options to up to five severity levels.

Analog Alarm Settings

Advanced Settings

Trigger Type: **Low** Alarm level(s) will activate when input is lower than trigger value. Trigger values must decrease with each level.

Clear Mode: **Auto** Once activated, alarm level(s) will remain latched until input is higher than the clear value.

Severity Levels

	Trigger	Clear	Syslog	SNMP
☒ Level 1	0.00000	0.00000	☐	☐
☐ Level 2	0.00000	0.00000	☐	☐
☐ Level 3	0.00000	0.00000	☐	☐
☐ Level 4	0.00000	0.00000	☐	☐
☐ Level 5	0.00000	0.00000	☐	☐

<< Basic OK Cancel Help

Configure the following parameters:

Trigger Type

If the **Trigger Type** is **Low**, an alarm is triggered when the input drops below the specified **Trigger** value; other severity level trigger values must decrease in value with each subsequent level. If the **Trigger Type** is **High**, an alarm is triggered when the input is higher than the specified **Trigger** value; other severity level trigger values must increase in value with each subsequent level.

Clear Mode

To clear an alarm, the input must drop below the specified value when **Trigger Type** is **High** or go above the specified value when **Trigger Type** is **Low**.

Level 1-5

Defines the Level severity settings for up to five levels. If the **Trigger Type** is **Low**, an alarm is triggered when the input drops below the specified **Trigger** value; other severity level trigger values must decrease in value with each subsequent level. If the **Trigger Type** is **High**, an alarm is triggered when the input is higher than the specified **Trigger** value; other severity level trigger values must increase in value with each subsequent level.

Trigger

If the **Trigger Type** is **Low**, an alarm is triggered when the input drops below the specified **Trigger** value; other severity level trigger values must decrease in value with each subsequent level. If the **Trigger Type** is **High**, an alarm is triggered when the input is higher than the specified **Trigger** value; other severity level trigger values must increase in value with each subsequent level.

Clear

To clear an alarm, the input must drop below the specified value when **Trigger Type** is **High** or go above the specified value when **Trigger Type** is **Low**.

Syslog

Sends a message to syslog when an alarm is triggered or cleared. The syslog entry includes the severity level and the value that caused the alarm to trigger or clear. The syslog message is associated with **Level Critical**.

SNMP

Sends an SNMP trap when an alarm is triggered or cleared. The trap consists of the severity level and whether the alarm was triggered or cleared.

Temperature

Temperature channels monitor either RTD or thermocouple inputs for the most common ranges.

Configure the following parameters:

- Enable Channel** Enables the channel, allowing the settings to become active.
- Description** Provide a description of the channel, making it easier to identify. The channel description can be up to 20 characters.
- Type** Specify the type of sensor you are using to measure temperature, either RTD or thermocouple. The default is RTD.
- Range** Specify the temperature range that you want to measure. For RTD, the default is Pt100 a=385 -50 to 150C. For thermocouple, the default is J 0 to 760C.

See [Basic Alarm Settings](#) on page 125 and [Advanced Alarm Settings](#) on page 126 for the Alarm Settings information.

Configuring Users

You can configure up to four users in the Device Server's local user database, in addition to the Admin user.

Configure the following parameters:

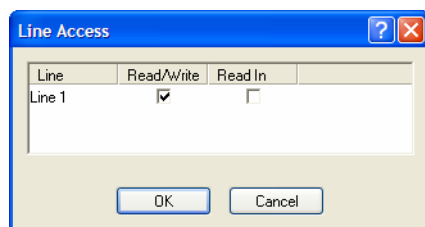
- User Name** The name of the user. Do not use spaces.
- Password** The password the user will need to enter to login to the Device Server.

Confirm Password	Enter the user's password again to verify it is entered correctly.
Level	The access that a user is allowed: • Admin—The admin level user has total access to the Device Server. You can create more than one admin user account but we recommend that you only have one. They can monitor and configure the Device Server. • Normal—The Normal level user has limited access to the Device Server. Limited CLI commands and Menu access are available with the ability to configure the user's own configuration settings. • Restricted—The Restricted level user can only access predefined sessions or access the Easy Port Access menu. • Menu—The menu level user will only be able to access predefined session or access the Easy Port Access menu. The Easy Port Access allows the user to connect to the accessible line without disconnecting their initial connection to the Device Server. Does not have any access to CLI commands.
Hotkey Prefix	The prefix that a user types to control the current session. The default value is hex 01, which corresponds to Ctrl-a (^a) (hex value 02 would be Ctrl-b (^b), etc.): • ^a number—To switch from one session to another, press ^a and then the required session number. For example, ^a 2 would switch you to session 2. Pressing ^a 0 will return you to the Device Server Menu. • ^a n—Display the next session. The current session will remain active. The lowest numbered active session will be displayed. • ^a p—Display the previous session. The current session will remain active. The highest numbered active session will be displayed. • ^a m—To exit a session and return to the Device Server. You will be returned to where you left off. The session will be left running. • ^a l—(Lowercase L) Locks the line until the user unlocks it. The user is prompted for a password (any password, excluding spaces) and locks the line. Next, the user must retype the password to unlock the line. • ^r—When you switch from a session back to the Menu, the screen may not be redrawn correctly. If this happens, use this command to redraw it properly. The User Hotkey Prefix value overrides the Line Hotkey Prefix value. You can use the Hotkey Prefix keys to lock a line only when the line Lock parameter is On.
Idle Timer	The amount of time, in seconds, that the Idle Timer will run. Use this timer to close a connection because of inactivity. When the Idle Timer expires, because there has been no exchange of data within the specified time, the Device Server will close the connection. The default value is 0 (zero), meaning that the Idle Timer will not expire (the connection is open permanently). The maximum value is 4294967 seconds. The User Idle Timer will override the Line Idle Timer, with the exception of reverse Telnet sessions.
Session Timer	The amount of time, in seconds, that the Session Timer will run. Use this timer to forcibly close a user's session (connection). When the Session Timer expires, the Device Server will end the connection. The default value is 0 (zero), meaning that the session timer will not expire (the session is open permanently, or until the user logs out). The maximum value is 4294967 seconds. The User Session Timer will override the Line Session Timer, with the exception of reverse Telnet sessions.

Language	You can specify whether a user will use English or Customlang as the language that appears in the Menu, CLI, or WebManager. The Device Server supports one custom language that must be downloaded to the Device Server; otherwise, Customlang defaults to English.
Service	The type of service that the user will use.
Host IP	When the User Service is set to Telnet or TCP_clear , the target host IP address. If 255.255.255.255 is specified in the configuration, the user will be prompted for an IP address or hostname. If no IP address is specified, the Host IP value in the Default User configuration will be used. The default is 0.0.0.0 .
TCP Port	When the User Service is Telnet , this is the target port number. The default value will change based on the type of Service selected; the most common known port numbers are used as the default values.

Configuring Line Access

Line Access defines the read/write privileges that a user has while accessing a line.



Configure the following options:

Line Access	Specifies the user access rights to each Device Server device line. Options are: • Read/Write—Users are given read and write access to the line. • Read In—Users are given access to read only outbound data, data that is going from the Device Server to the device.
--------------------	---

Configuring Sessions

You can configure user **Sessions** to limit the access the user has to the network and the way the user connects to a host. Users who are **Level Normal** or **Admin** can define **Free Sessions**, in addition to using defined sessions. Users who are **Level Restricted** or **Menu** can only access predefined sessions.

Configure the following parameters:

Session	You can create up to four predefined sessions for each user. You can specify the connection service and its settings for each session.
Auto	Specify whether or not the session(s) will start automatically when the user logs into the Device Server.

The following **Session** connections are available:

- **None**—No connection is configured for this session.
- **Telnet**—For information on the Telnet configuration window, see [Telnet Settings on page 104](#).

Configuring the Default User

When you add new users to the Device Server, they will initially inherit any parameters set in the **Default User** (the parameters can be changed on a per user basis).

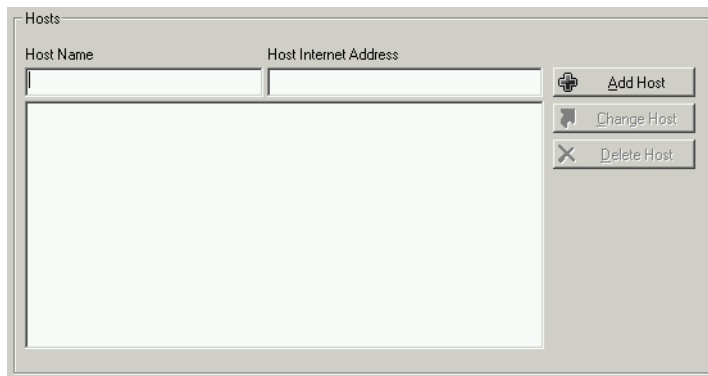
For information on the **Default User** configuration parameters, see [Configuring Users](#) on page 127.

Configuring the Network

The network configuration parameters define the network that the Device Server will be operating within.

Configuring Hosts

One of the first things you will probably want to configure is the hosts that the Device Server or Users will be interacting with, since most configuration windows require that the hosts already be configured. You can configure up to 20 hosts.

The screenshot shows a window titled "Hosts". At the top, there are two input fields: "Host Name" and "Host Internet Address". Below these fields is a large, empty rectangular area, likely a list or table for displaying configured hosts. To the right of this area are three buttons: "Add Host" (with a plus icon), "Change Host" (with a pencil icon), and "Delete Host" (with an X icon).

Configure the following parameters:

Host Name The name of the host.

Host Internet Address The host IP address.

Configuring SNMP

If you are using the Device Server SNMP MIB-based configuration/management option, you can use the DeviceManager to easily set up SNMP users, traps, and communities. The Device Server supports the SNMP traps for restart and SNMP community authentication error. For more information on SNMP, see [SNMP on page 54](#).

The image shows a graphical user interface for configuring SNMP. It is divided into several sections:

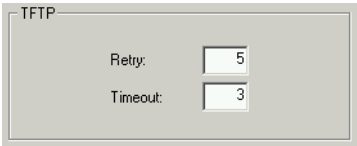
- Contact Information:** Contains two text input fields labeled 'Contact:' and 'Location:'.
- Communities (Version 1 and Version 2):** A table with three columns: 'Community', 'Internet Address', and 'Permissions'. There are four rows, each with an empty text field for the community name, an empty text field for the IP address, and a dropdown menu currently set to 'None'.
- Users (Version 3):** Contains two text input fields labeled 'Read-Write User:' and 'Read-Only User:'.
- Traps:** A table with two columns: 'Trap' and 'Internet Address'. There are four rows, each with an empty text field for the trap name and an empty text field for the IP address.

Configure the appropriate parameters:

Contact	The name and contract information of the person who manages this SNMP node.
Location	The physical location of the SNMP node.
Community	A name that will be sent to the Device Server from an SNMP manager. This name will define the permissions of the manager.
Internet Address	The IP address of the SNMP manager that will send requests to the Device Server. If the address is 0.0.0.0 , any SNMP manager with the Community Name can access the Device Server.
Permissions	Permits the Device Server to respond to SNMP requests by: • None—There is no response to requests from SNMP. • Readonly—Responds only to Read requests from SNMP. • Readwrite—Responds to both Read and Write requests from SNMP.
Read-Write User	Specified user can view and edit SNMP variables.
Read-Only User	Specified user can only view SNMP variables.
Trap	An arbitrary trap community name.
Internet Address	Defines the hosts (by IP address) that will receive trap messages generated by the Device Server. Up to four trap hosts can be defined.

Configuring TFTP

These parameters configure the TFTP settings for the Device Server’s connections to hosts (as opposed to the TFTP settings under **Tools, Options**, which configure the TFTP settings for the DeviceManager’s connection to a Device Server).

A screenshot of a configuration window titled "TFTP". It contains two settings: "Retry:" with a value of 5 and "Timeout:" with a value of 3. Both values are entered in small text boxes.

Configure the following parameters:

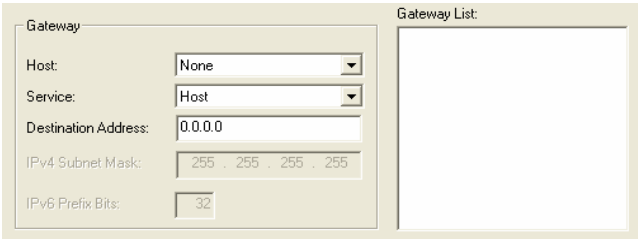
- Retry**

The number of times the Device Server will attempt to transfer (using TFTP) a file to/from a host. Enter a value between 0 and 5. The default is **5**. A value of **0** (zero) means that the Device Server will not attempt a retry should TFTP fail.
- Timeout**

The time, in seconds, that the Device Server will wait for a successful transmit or receipt of TFTP packets before retrying a TFTP transfer. Enter a value between 3 and 10. The default is **3** seconds.

Configuring Gateways

You can configure gateways to allow the Device Server access to hosts that are not within the local network segment.

A screenshot of a configuration window titled "Gateway". It has several fields: "Host:" with a dropdown menu showing "None", "Service:" with a dropdown menu showing "Host", "Destination Address:" with a text box containing "0.0.0.0", "IPv4 Subnet Mask:" with a text box containing "255 . 255 . 255 . 255", and "IPv6 Prefix Bits:" with a text box containing "32". To the right of these fields is a large empty box labeled "Gateway List:".

Configure the following parameters:

- Host**

You can specify up to twenty hosts to act as gateways in your network. Each gateway host must be defined in the Device Server host table.
- Service**

Specify the type of gateway:

 - **Default**—A gateway which provides general access beyond your local network.
 - **Host**—A gateway reserved for accessing a specific host external to your local network.
 - **Network**—A gateway reserved for accessing a specific network external to your local network.
- Destination Address**

When the gateway is a **Host** or **Network** gateway, you must specify the IP address of the target host machine/network.
- IPv4 Subnet Mask**

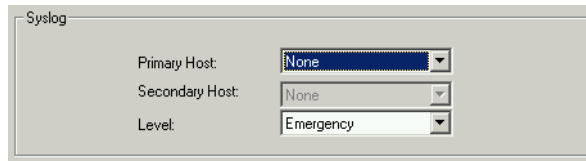
When the gateway is a **Network** gateway, you must specify the network’s subnet mask.
- IPv6 Prefix Bits**

If the IP address is IPv6, then the Prefix Bits range is 0-128.
- Gateway List**

The list of defined gateways.

Configuring Syslog

You can configure where the system log messages are going to be sent and specify the lowest level message that the Device Server will send syslog messages for.

A screenshot of a 'Syslog' configuration window. It contains three labels with corresponding dropdown menus: 'Primary Host' with 'None' selected, 'Secondary Host' with 'None' selected, and 'Level' with 'Emergency' selected.

Configure the following options:

- Primary Host** The first preconfigured host that the Device Server will attempt to send system log messages to; messages will be displayed on the host's monitor.
- Secondary Host** If the Device Server cannot communicate with the primary host, then the Device Server will attempt to send system log messages to this preconfigured host; messages will be displayed on the host's monitor.
- Level** Choose the event level that triggers a syslog entry:
- **Emergency**
 - **Alert**
 - **Critical**
 - **Error**
 - **Warning**
 - **Notice**
 - **Info**
 - **Debug**

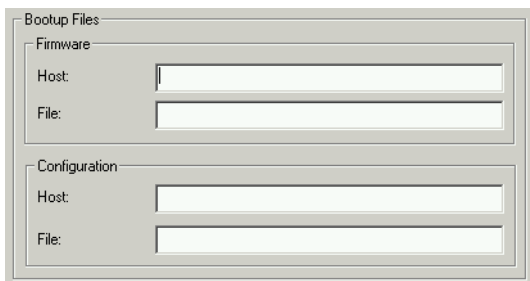
When you select a **Level**, all the levels that appear above it in the list also trigger a syslog entry. For example, if you select **Error**, all **Error**, **Critical**, **Alert**, and **Emergency** events will be logged.

Configuring Administration Tasks

You can specify new configuration and firmware files that will go into effect the next time the Device Server is rebooted and a message of the day (MOTD) file, whose contents will be displayed when User's log into the Device Server.

Configuring Bootup Files

When you specify a configuration and/or firmware file(s), the files will be downloaded via TFTP to the Device Server the next time it is rebooted.



The screenshot shows a window titled "Bootup Files". It is divided into two main sections: "Firmware" and "Configuration". Each section contains two text input fields: "Host" and "File".

Configure the following parameters:

- | | |
|---------------------------|--|
| Firmware Host | The host name or IP address of the server that contains the configuration or firmware file. If you use a host name, it must exist in the Device Server's host table or be resolved by DNS. |
| Firmware File | The path and file name (do not use a drive letter), relative to the default path of your TFTP server software, of the update software for the Device Server that will be loaded when the Device Server is rebooted. |
| Configuration Host | The host name or IP address of the server that contains the configuration or firmware file. If you use a host name, it must exist in the Device Server's host table or be resolved by DNS. |
| Configuration File | The path and file name (do not use a drive letter), relative to the default path of your TFTP server software, of the configuration software for the Device Server that will be loaded when the Device Server is rebooted. |

Configuring the MOTD File

You can specify a file whose content will be displayed to users after they connect to the Device Server, but before they log in. The Device Server will retrieve the file content every time a user connects to the Device Server, so you can change the content of the file without reconfiguring it within the Device Server.



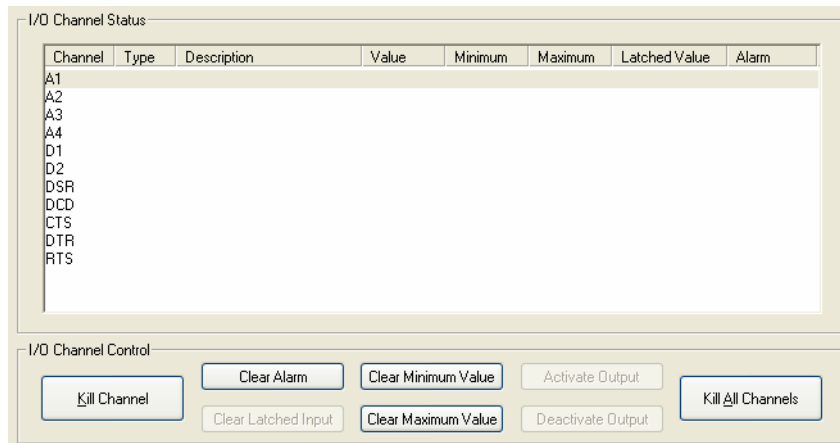
The screenshot shows a window titled "MOTD". It contains two text input fields: "Host" and "Filename".

Configure the following parameters:

- | | |
|-----------------|--|
| Host | The host that the Device Server will be getting the Message of the Day file from. |
| Filename | The path and file name (do not use a drive letter), relative to the default path of your TFTP server software, of the file that contains a string that is displayed when a user connects to the Device Server. |

I/O Status/Control

The I/O Status/Control window allow you to view I/O status and manually control I/O data.



A brief description of the control buttons follows:

- Kill Channel** Resets the highlighted channel (click on a channel to highlight it).
- Clear Alarm** Clears the alarm. Note that if the condition that tripped the alarm still exists, the alarm will not look like it's cleared, but will reflect the appropriate alarm level severity. Alarm Level 0 means that the alarm has not been triggered.
- Clear Latched Input** Clears the latch value.
- Clear Minimum Value** Clears the minimum value.
- Clear Maximum Value** Clears the maximum value.
- Activate Output** Manually activates the channel output.
- Deactivate Output** Manually deactivates the channel output.
- Kill All Channels** Resets all the channels.

Statistics

After you are connected to a Device Server, you can view statistics about the Device Server and its network environment. This can help you to troubleshoot problems or can provide valuable information about the Device Server's environment.

Tools

Saving a Configuration To File

When you connect to a Device Server, the Device Server's configuration is automatically uploaded to the DeviceManager. We suggest that you save the configuration to a file at this point, in case you need to revert to a working configuration in the future, by selecting **Tools, Save Configuration to File**. You can choose to save the configuration to the Device Server's native binary format or to a text file, which can be edited with a text editor. Either format can be reloaded into the DeviceManager at any time.

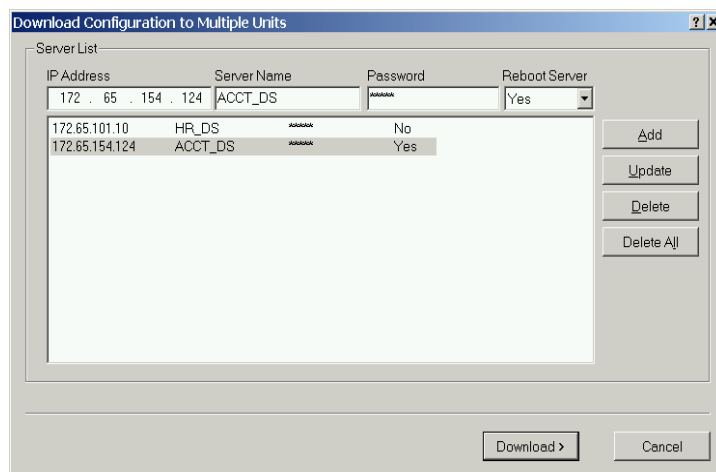
Getting a Configuration File

The DeviceManager can get a local configuration file (either binary or text) when you select **Tools, Get Configuration, Import from File**. The DeviceManager can also get the configuration from the Device Server it's connected to when you select **Tools, Get Configuration, Upload from Unit**; this can be useful if you've made changes to the Device Server's configuration that you would like to discard, you can simply reload the Device Server's current configuration into the DeviceManager.

Configuring Multiple Device Servers

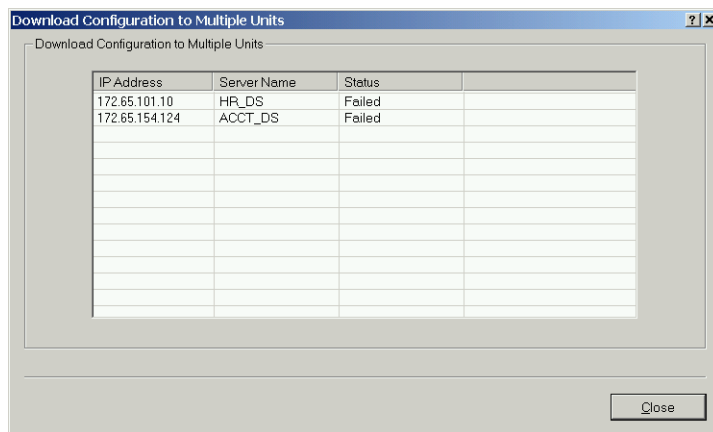
You can configure multiple Device Servers at one time with the active configuration file. Any value in the configuration file's **Server Name** and **Internet Address** parameters will be overwritten by the values specified in the **Server Name** and **IP Address** fields (these fields cannot be left blank)

1. Select **Tools, Download Configuration to Multiple Units**. The Download Configuration to Multiple Units window is displayed.



- | | |
|----------------------|---|
| IP Address | Enter the IP address of the Device Server that you want to download the configuration to. |
| Server Name | The name of the Device Server. The Device Server name that you put in this field is passed into the configuration before it is downloaded to the Device Server and cannot be left blank. |
| Password | Enter the Admin user password for the Device Server. |
| Reboot Server | Determines whether or not the Device Server is rebooted after it has received the new configuration. The new configuration definitions will not go into effect until the Device Server is rebooted. |

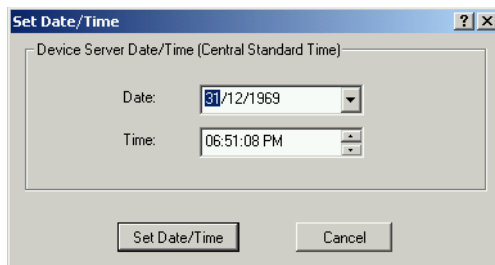
- Click **Add** to add the Device Server to the download list. You can also click on a Device Server and edit any information and then click **Update** to make the edits permanent.
- Click the **Download>** button to start the download process. A status window will display with the configuration download status.



To upgrade the Device Server firmware (software), select **Tools, Download Firmware to Unit**. Once the firmware download is complete, you will be prompted to reboot the Device Server. You can choose to reboot the Device Server at another time by selecting **Tools, Reboot Server**. Upgrading the firmware does not affect the Device Server's configuration file or downloaded custom files.

Setting the Device Server's Date and Time

To set the Device Server's system clock, select **Tools, Set Unit Time/Date**. The Set Date/Time window is displayed.



Configure the following parameters:

- | | |
|-------------|---|
| Date | The Device Server's date. The format of the Device Server's date is dependent on the Windows operating system and regional settings. |
| Time | The Device Server's internal clock time, based on your PC's time zone. For example, if your PC's time zone is set to Pacific Standard Time (GMT -8:00) and the Device Server's time zone is set to Eastern Standard Time (GMT -5:00), the Device Server's time is three hours ahead of your PC's time. If you set the Device Server's time to 2:30 pm, the Device Server's actual internal clock time is 5:30 pm. |

Rebooting the Device Server

When you download any file (configuration, keys, certificates, firmware, etc.) to the Device Server, you must reboot the Device Server for it to take effect by selecting **Tools, Reboot Server**.

Resetting the Device Server to Factory Defaults

You can reset the Device Server to its factory default configuration by selecting **Tools, Reset to Factory Default**. The Device Server will automatically reboot itself with the factory default configuration.

Resetting a Line

After you make changes to the **Line** configuration parameters and click the **Apply** button, you can reset the line to test the changes by selecting **Tools, Kill Line**. If you are happy with the configuration changes, you can download the configuration by selecting **Tools, Download Configuration to Unit**. Of course, your new configuration will not take effect until you reboot the Device Server by selecting **Tools, Reboot Server**.

Custom Files

Saving Crashes to a Dump File

If the Device Server should crash, you can save the crash information (dump) to a file that can be sent to Technical Support for interpretation. This should probably be done only under the guidance of Technical Support.

Downloading Terminal Definitions

You can create up to three custom terminal definitions and download them to the Device Server (if you need a terminal definition that is not currently defined within the Device Server). It is important that you remember which Device Server Terminal Definition you download your custom terminal definition under.

For example, if you download a custom terminal definition as **Terminal Definition 2**, you must select **Terminal Type Term2** in the **Line** window to use that terminal definition.

See [Creating Terminal Definition Files on page 71](#) for information on creating custom terminal definitions.

Downloading a Language File

You can download one custom language file that can be specified in the **User** configuration window. See [Language support on page 69](#) for information on creating custom language files.

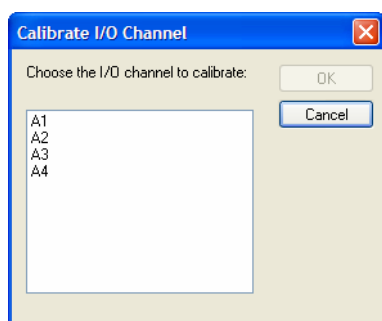
Downloading a Custom App/Plugin File

You can download each custom application file, created with the Perle SDK, to the Device Server using this option.

I/O Channels

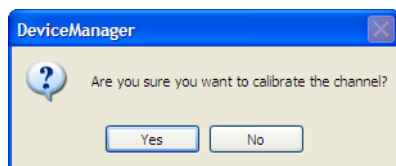
Calibrating Analog Channels

Analog Input can be calibrated for Analog and Temperature Device Server models. To calibrate either of these models, select **Tools, I/O Channels, Calibrate**.

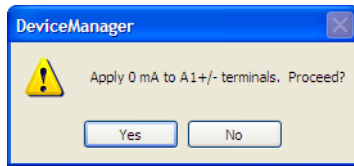


Select the channel you want to calibrate. This example uses an A4 model that has channel A1 set to Current with a Range of 0 to 20mA.

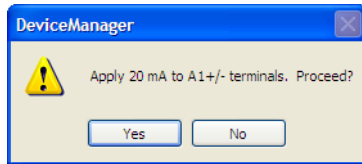
If you have not disabled confirmation messages (**Tools, Options**), you will get prompted to verify channel calibration.



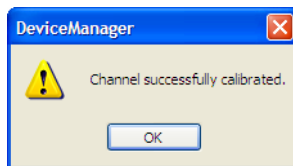
Click **Yes** to proceed with calibration. You are now prompted to apply 0 mA to the positive (+) and negative (-) terminals. Once that is done, click **Yes** to proceed.



You are now prompted to apply 20 mA to the positive (+) and negative (-) terminals. Once that is done, click **Yes** to proceed.



Once calibration is successfully completed, click **OK** to finish the process.



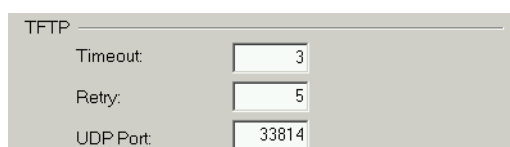
Resetting Calibration Data

When you select **Tools, I/O Channels, Reset Calibration Data**, you are resetting the calibration values to the factory default values.

Setting DeviceManager Options

When you select **Tools, Options**, you can set the following:

- **Confirmation Messages**—Specify whether you want to receive confirmation messages for all of the following selections:
 - **Tools, Download Configuration to Unit**
 - **Tools, Reboot Server**
 - **Tools, Reset to Factory Defaults**
 - **Tools, Reset SecuriD Node Secret**
 - **Tools, Kill Line**
 - Anytime you click a **Delete** button
- **TFTP**—Sets the TFTP options for communication between the DeviceManager and a Device Server.



TFTP	
Timeout	3
Retry	5
UDP Port	33814

Configure the following parameters:

Retry The number of times the DeviceManager will attempt to transfer (using TFTP) a file to/from a Device Server. Enter a value between 0 and 5. The default is **5**. A value of **0** (zero) means that the Device Server will not attempt a retry should TFTP fail.

Timeout The time, in seconds, that the DeviceManager will wait for a successful transmit or receipt of TFTP packets before retrying a TFTP transfer. Enter a value between 3 and 10. The default is **3** seconds.

UDP Port The port that the DeviceManager will use to TFTP to the Device Server. The default port is 33814 (ports 33812 and 33813 are also in use by the DeviceManager).

- **Statistics**—Specify whether or not you want to have the statistics automatically refresh and the refresh rate.



Command Line Interface

Introduction

This chapter provides the command line interface (CLI) options available for the Device Server. The commands are grouped by function.

CLI Conventions

This section explains how to interpret the CLI syntax.

Command Syntax

Each command is broken down into several categories:

- **Description**—Provides a brief explanation of how the command is used.
- **User Level**—Shows which user level(s) (Restricted, Normal, and/or Admin) can issue the command. Some commands have options that are available for one user level and not for another level; this usually occurs when a command is valid for both Normal and Admin user levels, where the Admin user level command will have extended options.
- **Syntax**—Shows the actual command line options. The options can be typed in any order on the command line. The syntax explanation will use the following command to break down the command syntax:

```
set service [dhcp/bootp on|off] [telnetd on|off] [httpd on|off]  
[snmpd on|off] [spcd on|off] [syslog on|off] [dmgrd on|off]
```

 - Square brackets ([]) show the options that are available for the command. You can type a command with each option individually, or string options together in any order you want. For example,
set service dhcp/bootp on telnetd off
 - Angle brackets (<>) show that the text inside the brackets is a description for a variable value that you must fill in according to your requirements. In the **set server** command, you must determine the values for **domain**, **internet**, **name**, **password-limit**, and **subnet-bit-length**, if you wish to specify them and not use their defaults (default values provided in the **Options** description). The angle brackets can also contain a range that can be used.
 - The pipe (|) shows an 'or' condition. For example, valid values for **telnetd** are either **on** or **off**.
- **Options**—Provides an explanation of each of the options for a command and the default value if there is one. Some commands do not have any options, so this category is absent.

Command Shortcuts

When you type a command, you can specify the shortest unique version of that command or you can press the **ESC** or **TAB** key to complete the command. For example, the following command:

```
set telnet-client map-to-crlf off
```

can be typed as:

```
set tel map off
```

or, you can use the **ESC** key to complete the lines as you go along:

```
set tel<ESC>net-client ma<ESC>p-to-crlf off
```

where the <ESC> key was pressed to complete the option as it was typed.

Command Options

When you are typing commands on the command line (while connected to the Device Server), you can view the options by typing a question mark (?) after any part of the command to see what options are available/valid. For example:

```
DS$ set vmodem ?
failure-string
host
port
style
success-string
suppress
DS$ set vmodem failure-string ?
<text>                 30 characters maximum
DS$ set vmodem failure-string "Vmodem failed" ?
failure-string
host
port
style
success-string
suppress
Or press Enter to confirm command
DS$ set vmodem failure-string "Vmodem failed"
DS$ show vmodem
Host
Host Port
Success String
Failure String          "Vmodem failed"
Suppress                Off
Style                   Numeric
DS$
```

Server Commands

This section defines all the CLI commands associated with configuring the Device Server's server parameters.

Server Commands

Set Custom-App

Description You can create a custom application that can run on the Device Server using the Perle SDK.

User Level Admin

Syntax **set custom-app server program-command-line** *<command>*

Options **program-command-line**

The name of the SDK program executable that has been already been downloaded to the Device Server, plus any parameters you want to pass to the program. Maximum of 80 characters. Use the **shell** CLI command as described in the *SDK Programmer's Guide* to manage the files that you have downloaded to the Device Server. For example, using sample outraw program, you would type:

```
outraw -s 0 192.168.2.1:10001 Acct:10001
```

if you were starting the application on the Server (notice the **-s 0** parameter specifies Line 1).

Set Server

Description Sets server parameters.

User Level Admin

Syntax **set server** [**banner on|off**] [**bypass-password on|off**] [**domain** *<string>*] [**flush-on-close on|off**] [**internet** *<IPV4_address>*] [**name** *<string>*] [**oem-login on|off**] [**password-limit** *<0-10>*] [**prompt-with-name on|off**] [**single-telnet on|off**] [**netmask** *<IPV4_address>*]

set server tftp [**retry** *<integer>*] [**timeout** *<integer>*]

Options **banner**

This parameter concerns the banner information (product name/software version). This banner information is presented to a user with a login prompt. For security reasons, you can turn off the display of this information. The default is **Off**.

bypass-password

When set, authorised users who do not have a password set, with the exception of the Admin user, WILL NOT be prompted for a password at login with **Local Authentication**.

domain_name

Unique name for your domain, your location in the global network. Like Hostname, it is a symbolic, rather than a numerical, identifier.

flush-on-close

When enabled, deletes any pending data when a port is closed; as opposed to maintaining the port to send pending data. The default value is **Off**.

internet

The Device Server's unique IPv4 network IP address. If you are using the Device Server in an IPv6 network, this field can be left blank.

oem-login

When set, and a custom language file is in use, the login prompt will use the string defined in the language file as the login prompt instead of the default prompt, **login:.**

password-limit

The number of attempts a user is allowed to enter a password for a port. If this limit is exceeded, the port is disabled for 5 minutes. A user with Admin level rights can restart the port, bypassing the timeout, by issuing a kill on the disabled port. The default value is **3**.

prompt-with-name

Displays the **Server Name** field value instead of default product name. When enabled, the **Server Name** is displayed in the Device Server login prompt, CLI prompt, WebManager login screen, and the heading of the Menu. The default value is **Off**.

single-telnet

Sets all reverse connections (raw and telnet) to a one connection at a time mode. Server-side applications will get a (socket) connection refused until:

- All data from previous connections on that serial port has drained
- There are no other connections
- Up to a 1 second interconnection poll timer has expired

This also enables a per-connection keepalive TCP keepalive feature. After approximately 3 minutes of network connection idle time, the connection will send a gratuitous ACK to the network peer, thus either ensuring the connection stays active OR causing a dropped connection condition to be recognised by the reverse service (all connections).

Applications using Single Telnet need to be aware that there can be some considerable delay between a network disconnection and the port being available for the next connection attempt; this is to allow any data sent on prior connections to be transmitted out of the serial port. Application network retry logic needs to accommodate this feature. The default value is **Off**.

netmask

The network subnet mask. For example, 255.255.0.0.

retry

The number of times the Device Server will attempt to transfer (using TFTP) a file to/from a host. Enter a value between 0 and 5. The default is **5**. A value of **0** (zero) means that the Device Server will not attempt a retry should TFTP fail.

timeout

The time, in seconds, that the Device Server will wait for a successful transmit or receipt of TFTP packets before retrying a TFTP transfer. Enter a value between 3 and 10. The default is **3** seconds.

Set Service

Description Sets server service parameters.

User Level Admin

Syntax **set service** [dhcp/bootp on|off] [telnetd on|off] [httpd on|off] [snmpd on|off] [spcd on|off] [syslog on|off] [dmgrd on|off] [modbusd on|off]

Options **dhcp/bootp**

DHCP/BOOTP client process in the Device Server. By default, this is disabled/off. If this is enabled, the server IP address parameter is disabled.

telnetd

Telnet daemon process in the Device Server on port 23.

httpd

HTTP daemon process in the Device Server on port 80.

snmpd

SNMP daemon process in the Device Server on port 161.

spcd

SPC (Trueport) daemon process in the Device Server on port 688.

syslog

Syslog client process in the Device Server.

dmgrd

DeviceManager daemon process in the Device Server. If you disable this service, you will not be able to connect to the Device Server with the DeviceManager application. DeviceManagerD listens on port 33812 and sends on port 33813.

modbusd

Modbus daemon process in the Device Server on port 502.

Show Custom-App

Description Shows the custom application server settings.

User Level Admin

Syntax **show custom-app server**

Show Modbus

Description Shows the Modbus settings for the gateway.

User Level Normal, Admin

Syntax **show modbus gateway**

Show Server

Description Shows the parameters set for the server.

User Level Admin, Normal

Syntax **show server**

Hardware Commands

Set Ethernet

Description Sets the serial line speed and duplex.

User Level Admin

Syntax `set ethernet speed-and-duplex
auto|10-half|10-full|100-half|100-full`

Options `auto|10-half|10-full|100-half|100-full`

Define the ethernet connection speed at one of the following:

- **auto**—automatically detects the ethernet interface speed and duplex
- **10 Mbps Half Duplex**
- **10 Mbps Full Duplex**
- **100 Mbps Half Duplex**
- **100 Mbps Full Duplex**

Show Hardware

Description Shows the hardware settings/information.

User Level Normal, Admin

Syntax `show hardware`

Modbus Commands

Set Modbus Gateway

Description Sets the authentication method for the Device Server.

User Level Admin

Syntax `set modbus gateway [addr-mode embedded|re-mapped]
[broadcast on|off] [char-timeout <number>]
[req-next-delay <number>] [exceptions on|off]
[idle-timer <number>] [mess-timeout <number>]
[port <TCP/UDP_port>] [req-queuing on|off]
[remapped-id <1-247>] [ssl on|off]`

Options `addr-mode`

Determines if the original UID address will be embedded in the transmission header or if a specified (remapped) UID will be embedded in the transmission header.

broadcast

When enabled, a UID of 0 (zero) indicates that the message will be broadcast to all Modbus Slaves. The default is **Off**.

char-timeout

Used in conjunction with the Modbus RTU protocol, specifies how long to wait, in milliseconds, after a character to determine the end of frame. The default is **30** ms.

req-next-delay

A delay, in milliseconds, to allow serial slave(s) to re-enable receivers before issuing next Modbus Master request. The default is **50** ms.

exceptions

When enabled, an exception message is generated and sent to the initiating Modbus device when any of the following conditions are encountered: there is an invalid UID, the UID is not configured in the Gateway, there is no free network connection, there is an invalid message, or the target device is not answering the connection attempt. The default is **On**.

idle-timer

Specifies the number of seconds that must elapse without any network or serial traffic before a connection is dropped. If this parameter is set to 0 (zero), a connection will not be dropped (with the following exceptions: the TCP KeepAlive causes the connection to be dropped or the Modbus device drops the connection). The default is **10** seconds.

mess-timeout

Time to wait, in milliseconds, for a response message from a Modbus TCP or serial slave (depending if the Modbus Gateway is a Master Gateway or Slave Gateway, respectively) before sending a Modbus exception. The default is **1000** ms.

port

The network port number that the Slave Gateway will listen on for both TCP and UDP messages. The default is **502**.

req-queuing

When enabled, allows multiple, simultaneous messages to be queued and processed in order of reception. The default is **On**.

remapped-id

Specify the UID that will be inserted into the message header for the Slave Modbus serial device. Valid values are 1-247.

ssl

When enabled, Modbus Slave Gateway messages to remote TCP Modbus Masters are encrypted via SSL/TLS.

Show Modbus

Description Sets the authentication method for the Device Server.

User Level Admin

Syntax **show modbus gateway**

show modbus slave|master <line_number>

TruePort Baud Commands

Set TruePort Remap-Baud

Description Sets the TruePort baud remapping values.

User Level Admin

Syntax **set trueport remap-baud**

50|75|110|134|150|200|300|600|1200|1800|2400|4800|9600|19200|38400

50|75|110|134|150|200|300|600|1200|1800|2400|4800|9600|19200|38400|57600|115200|230400|28800|[custom <baud_rate>]

Options **50|75|110|134|150|200|300|600|1200|1800|2400|4800|9600|19200|38400**

The configured baud rate of the TruePort client.

50|75|110|134|150|200|300|600|1200|1800|2400|4800|9600|19200|38400|57600|115200|230400|28800|[custom <baud_rate>]

The actual baud rate that runs between the Device Server and the connected serial device. You can also specify a custom baud rate; valid values are 50-230400.

Show TruePort

Description Shows the Device Server TruePort remapping table.

User Level Normal, Admin

Syntax **show trueport**

User Commands

Logged Into the Device Server Commands

Admin

Description Changes a Normal-level user to the Admin user. When you press **Enter** after you type this command, you will be prompted for the Admin password.

User Level Normal

Syntax **admin**

Help

Description Displays help on using the command line interface (CLI).

User Level Restricted, Normal, Admin

Syntax **help**

Kill Line

Description Restarts a line.

User Level Normal, Admin

Syntax **kill line**

Kill Session

Description Kills an active session.

User Level Restricted, Normal, Admin

Syntax **kill session 1|2|3|4**

Options **1|2|3|4**

The number of the session(s) you want to kill.

Logout

Description Logs the user out from the Device Server.

User Level Restricted, Normal, Admin

Syntax **logout**

Menu

Description Switches from the CLI mode to the Menu.

User Level Restricted, Normal, Admin

Syntax **menu**

Ping

Description Pings the specified host/IP address.

User Level Normal, Admin

Syntax **ping** *<hostname/IP_address>* [*<packet_size>*] [*<#_of_packets>*]

Options *<hostname/IP_address>*

The name (host name or DNS name) or IP address of the machine you are trying to ping (verify the connection with).

<packet_size>

Enter the number of data bytes to be sent. The maximum size is 2000 bytes.

<#_of_packets>

Enter the number of the packets you want to send.

Resume

Description Resumes a started session.

User Level Restricted, Normal, Admin

Syntax **resume** **1|2|3|4**

Options **1|2|3|4**

The number of the session you want to resume.

Screen

Description Switches from the CLI mode to the Menu.

User Level Restricted, Normal, Admin

Syntax **screen**

Set Termttype

Description Sets the type of terminal being used for the current session.

User Level Normal, Admin

Syntax **set termttype**
wyse60|vt100|ansi|dumb|tvi925|ibm3151te|vt320|hp700|term1|term2|term3

Option **wyse60|vt100|ansi|dumb|tvi925|ibm3151te|vt320|hp700|term1|term2|term3**

Specifies the type of terminal connected to the line:

- **Dumb**
- **WYSE60**
- **VT100**
- **ANSI**
- **TVI925**
- **IBM3151TE**
- **VT320** (specifically supporting VT320-7)
- **HP700** (specifically supporting HP700/44)
- **Term1, Term2, Term3** (user defined terminals)

Set User

Description Sets the current user's settings.

User Level Normal, Admin

Syntax **set user . [hotkey-prefix <00-7f>] [language english|customlang] [password]**

Options **hotkey-prefix**

The prefix that a user types to control the current session. The default value is **hex 01**, which corresponds to **Ctrl-a (^a)** (hex value 02 would be Ctrl-b (^b), etc.):

- **^a number**—To switch from one session to another, press **^a** and then the required session number. For example, **^a 2** would switch you to session 2. Pressing **^a 0** will return you to the Device Server Menu.
- **^a n**—Display the next session. The current session will remain active. The lowest numbered active session will be displayed.
- **^a p**—Display the previous session. The current session will remain active. The highest numbered active session will be displayed.
- **^a m**—To exit a session and return to the Device Server. You will be returned to where you left off. The session will be left running.
- **^a l**—(Lowercase L) Locks the line until the user unlocks it. The user is prompted for a password (any password, excluding spaces) and locks the line. Next, the user must retype the password to unlock the line.
- **^r**—When you switch from a session back to the Menu, the screen may not be redrawn correctly. If this happens, use this command to redraw it properly.

The **User Hotkey Prefix** value overrides the **Line Hotkey Prefix** value. You can use the **Hotkey Prefix** keys to lock a line only when the line **Lock** parameter is **On**.

language

You can specify whether a user will use **English** or **Customlang** as the language that appears in the Menu, CLI, or WebManager. The Device Server supports one custom language that must be downloaded to the Device Server; otherwise, **Customlang** defaults to English.

password

The password the user will need to enter to login to the Device Server.

Set User Session

Description Sets the current user's session settings.

User Level Normal, Admin

Syntax `set user . session 1|2|3|4|* [auto on|off] [type off|telnet]`

```
set user . session 1|2|3|4|* telnet-options [host <config_host>]
[port <TCP_port>] [termttype <terminal_name>] [line-mode on|off]
[map-cr-crlf on|off] [local-echo on|off] [echo <00-7f>]
[eof <00-7f>] [erase <00-7f>] [intr <00-7f>] [quit <00-7f>]
```

Options `session`

Specifies the session number (or all, *) that you are configuring.

`auto`

Specify whether or not the session(s) will start automatically when the user logs into the Device Server.

`telnet-options`

See [Set Telnet-Client](#) on page 168.

Show Line Users

Description Shows the users who are on the line.

User Level Admin

Syntax `show line users`

Syslog Console

Description Starts/stops or displays the status of the syslog console.

User Level Admin

Syntax `syslog console start|stop`

`syslog console status`

Options `start|stop`

Start or stop console logging. When console logging is enabled, syslog messages will be echoed to the current console. These messages are filtered based on the level set in the (remote) syslog options.

`status`

Displays the current console logging status (enabled or disabled).

Show Sessions

Description Shows available sessions.

User Level Restricted, Normal, Admin

Syntax `show sessions`

Show Termttype

Description Shows the terminal type for the current session.

User Level Admin

Syntax `show termttype`

Start

Description Starts a predefined session. Only inactive sessions are displayed.

User Level Restricted, Normal, Admin

Syntax **start** 1|2|3|4

Options 1|2|3|4

The number of the session that you want to start.

Telnet

Description Starts a telnet session to the specified host/IP address.

User Level Normal, Admin

Syntax **telnet** <hostname/IP_address> [<tcp_port>]
[termtype <terminal_name>] [line-mode on|off]
[map-cr-crlf on|off] [local-echo on|off]
[echo <00-7f>] [eof <00-7f>] [erase <00-7f>] [intr <00-7f>]
[quit <00-7f>] [escape <00-7f>]

Options <hostname/IP_address>

The name of the target host.

<tcp_port>

The port number the target host is listening on for incoming connections.

termtype

Type of terminal attached to this line; for example, ANSI or WYSE60.

line-mode

When **On**, keyboard input is not sent to the remote host until **Enter** is pressed, otherwise input is sent every time a key is pressed. Default is **Off**.

map-cr-crlf

Maps carriage returns (CR) to carriage return line feed (CRLF). The default value is **Off**.

local-echo

Toggles between local echo of entered characters and suppressing local echo. Local echo is used for normal processing, while suppressing the echo is convenient for entering text that should not be displayed on the screen, such as passwords. This parameter can only be used when **Line Mode** is **On**. Default is **Off**.

echo

Defines the echo character. When Line Mode is On, typing the echo character echoes the text locally and sends only completed lines to the host. This value is in hexadecimal with a default value of **5** (ASCII value **^E**).

eof

Defines the end-of-file character. When Line Mode is On, entering the eof character as the first character on a line sends the character to the remote host. This value is in hexadecimal with a default value of **4** (ASCII value **^D**).

erase

Defines the erase character. When Line Mode is Off, typing the erase character erases one character. This value is in hexadecimal with a default value of **8** (ASCII value **^H**).

intr

Defines the interrupt character. Typing the interrupt character interrupts the current process. This value is in hexadecimal with a default value of **3** (ASCII value **^C**).

quit

Defines the quit character. Typing the quit character closes and exits the current telnet session. This value is in hexadecimal with a default value of **1c** (ASCII value **FS**).

escape

Defines the escape character. Returns you to the command line mode. This value is in hexadecimal with a default value of **1d** (ASCII value **GS**).

Version

Description Displays firmware version and build.

User Level Normal, Admin

Syntax **version**

Configuring Users

Add User

Description Adds a user. You can add and configure up to four users in the Device Server.

User Level Admin

Syntax **add user** *<username>*

Option *<username>*

The name of the user, without spaces. When you finish the command and press Enter, you will be prompted to enter and re-enter a password for the user.

Delete User

Description Deletes a user.

User Level Admin

Syntax **delete user** *<config_user>*

Option *<config_user>*

You can see a list of users that can be deleted by typing **delete user ?**.

Set Default User

Description Configures the Default User.

User Level Admin

Syntax `set default user [hotkey-prefix <00-7f>]
[idle-timer <0-4294967>] [ip-host <ip_address>]
[language english|customlang]
[level admin|normal|restricted|menu]
[line-access readin|readwrite on|off]
[service dsprompt|telnet|tcp-clear] [sess-timer <0-4294967>]
[port tcp-clear|telnet <tcp_port>]`

Options **hotkey-prefix**

The prefix that a user types to control the current session. The default value is **hex 01**, which corresponds to **Ctrl-a (^a)** (hex value 02 would be Ctrl-b (^b), etc.):

- **^a number**—To switch from one session to another, press **^a** and then the required session number. For example, **^a 2** would switch you to session 2. Pressing **^a 0** will return you to the Device Server Menu.
- **^a n**—Display the next session. The current session will remain active. The lowest numbered active session will be displayed.
- **^a p**—Display the previous session. The current session will remain active. The highest numbered active session will be displayed.
- **^a m**—To exit a session and return to the Device Server. You will be returned to where you left off. The session will be left running.
- **^a l**—(Lowercase L) Locks the line until the user unlocks it. The user is prompted for a password (any password, excluding spaces) and locks the line. Next, the user must retype the password to unlock the line.
- **^r**—When you switch from a session back to the Menu, the screen may not be redrawn correctly. If this happens, use this command to redraw it properly.

The **User Hotkey Prefix** value overrides the **Line Hotkey Prefix** value. You can use the **Hotkey Prefix** keys to lock a line only when the line **Lock** parameter is **On**.

idle-timer

The amount of time, in seconds, that the **Idle Timer** will run. Use this timer to close a connection because of inactivity. When the **Idle Timer** expires, because there has been no exchange of data within the specified time, the Device Server will close the connection. The default value is **0** (zero), meaning that the **Idle Timer** will not expire (the connection is open permanently). The maximum value is 4294967 seconds. The **User Idle Timer** will override the **Line Idle Timer**, with the exception of reverse Telnet sessions.

ip-host

When the **User Service** is set to **Telnet** or **TCP_clear**, the target host IP address. If 255.255.255.255 is specified in the configuration, the user will be prompted for an IP address or hostname. If no IP address is specified, the Host IP value in the **Default User** configuration will be used. The default is **0.0.0.0**.

language

You can specify whether a user will use **English** or **Customlang** as the language that appears in the Menu, CLI, or WebManager. The Device Server supports one custom language that must be downloaded to the Device Server; otherwise, **Customlang** defaults to English.

level

The access that a user is allowed:

- **Admin**—The admin level user has total access to the Device Server. You can create more than one admin user account but we recommend that you only have one. They can monitor and configure the Device Server.
- **Normal**—The Normal level user has limited access to the Device Server. Limited CLI commands and Menu access are available with the ability to configure the user's own configuration settings.
- **Restricted**—The Restricted level user can only access predefined sessions or access the Easy Port Access menu.
- **Menu**—The menu level user will only be able to access predefined session or access the Easy Port Access menu. The Easy Port Access allows the user to connect to the accessible line without disconnecting their initial connection to the Device Server. Does not have any access to CLI commands.

password

The password the user will need to enter to login to the Device Server.

line-access

Specifies the user access rights to each Device Server device line. Options are:

- **Read/Write**—Users are given read and write access to the line.
- **Read In**—Users are given access to read only outbound data, data that is going from the Device Server to the device.

phone-number

The phone number the Device Server will dial to callback the user (you must have set **Callback to On**). Enter the number without spaces. To change the phone number, overwrite the previous entry; to clear the phone number, set it to "" (double quotes without a space).

service

The type of service that the user will use.

sess-timer

The amount of time, in seconds, that the **Session Timer** will run. Use this timer to forcibly close a user's session (connection). When the **Session Timer** expires, the Device Server will end the connection. The default value is **0** (zero), meaning that the session timer will not expire (the session is open permanently, or until the user logs out). The maximum value is 4294967 seconds. The **User Session Timer** will override the **Line Session Timer**, with the exception of reverse Telnet sessions.

port

When the **User Service** is **Telnet**, this is the target port number. The default value will change based on the type of **Service** selected; the most common known port numbers are used as the default values.

Set User

Description	Sets user's settings. Normal-level users can configure only their own settings. Admin-level users can configure any user's settings, including their own (with the exception of their User Level, which must stay at Admin).
User Level	Normal, Admin
Syntax	set user . [hotkey-prefix <00-7f>] [language english customlang] [password]
Admin	set user . <username> * [hotkey-prefix <00-7f>] [idle-timer <0-4294967>] [ip-host <ip_address>] [language english customlang] [level admin normal restricted menu] [password] [line-access readin readout readwrite on off] [service dsprompt telnet tcp-clear] [sess-timer <0-4294967>] [port tcp-clear telnet <tcp_port>]
Options	hotkey-prefix

The prefix that a user types to control the current session. The default value is **hex 01**, which corresponds to **Ctrl-a (^a)** (hex value 02 would be Ctrl-b (^b), etc.):

- **^a number**—To switch from one session to another, press **^a** and then the required session number. For example, **^a 2** would switch you to session 2. Pressing **^a 0** will return you to the Device Server Menu.
- **^a n**—Display the next session. The current session will remain active. The lowest numbered active session will be displayed.
- **^a p**—Display the previous session. The current session will remain active. The highest numbered active session will be displayed.
- **^a m**—To exit a session and return to the Device Server. You will be returned to where you left off. The session will be left running.
- **^a l**—(Lowercase L) Locks the line until the user unlocks it. The user is prompted for a password (any password, excluding spaces) and locks the line. Next, the user must retype the password to unlock the line.
- **^r**—When you switch from a session back to the Menu, the screen may not be redrawn correctly. If this happens, use this command to redraw it properly.

The **User Hotkey Prefix** value overrides the **Line Hotkey Prefix** value. You can use the **Hotkey Prefix** keys to lock a line only when the line **Lock** parameter is **On**.

idle-timer

The amount of time, in seconds, that the **Idle Timer** will run. Use this timer to close a connection because of inactivity. When the **Idle Timer** expires, because there has been no exchange of data within the specified time, the Device Server will close the connection. The default value is **0** (zero), meaning that the **Idle Timer** will not expire (the connection is open permanently). The maximum value is 4294967 seconds. The **User Idle Timer** will override the **Line Idle Timer**, with the exception of reverse Telnet sessions.

ip-host

When the **User Service** is set to **Telnet** or **TCP_clear**, the target host IP address. If 255.255.255.255 is specified in the configuration, the user will be prompted for an IP address or hostname. If no IP address is specified, the Host IP value in the **Default User** configuration will be used. The default is **0.0.0.0**.

language

You can specify whether a user will use **English** or **Customlang** as the language that appears in the Menu, CLI, or WebManager. The Device Server supports one custom language that must be downloaded to the Device Server; otherwise, **Customlang** defaults to English.

level

The access that a user is allowed:

- **Admin**—The admin level user has total access to the Device Server. You can create more than one admin user account but we recommend that you only have one. They can monitor and configure the Device Server.
- **Normal**—The Normal level user has limited access to the Device Server. Limited CLI commands and Menu access are available with the ability to configure the user's own configuration settings.
- **Restricted**—The Restricted level user can only access predefined sessions or access the Easy Port Access menu.
- **Menu**—The menu level user will only be able to access predefined session or access the Easy Port Access menu. The Easy Port Access allows the user to connect to the accessible line without disconnecting their initial connection to the Device Server. Does not have any access to CLI commands.

password

The password the user will need to enter to login to the Device Server.

line-access

Specifies the user access rights to each Device Server device line. Options are:

- **Read/Write**—Users are given read and write access to the line.
- **Read In**—Users are given access to read only outbound data, data that is going from the Device Server to the device.

service

The type of service that the user will use.

sess-timer

The amount of time, in seconds, that the **Session Timer** will run. Use this timer to forcibly close a user's session (connection). When the **Session Timer** expires, the Device Server will end the connection. The default value is **0** (zero), meaning that the session timer will not expire (the session is open permanently, or until the user logs out). The maximum value is 4294967 seconds. The **User Session Timer** will override the **Line Session Timer**, with the exception of reverse Telnet sessions.

port

When the **User Service** is **Telnet**, this is the target port number. The default value will change based on the type of **Service** selected; the most common known port numbers are used as the default values.

Set User Session

Description Configures a user's session settings. See [Set User Session on page 153](#) for the options descriptions.

User Level Admin

Syntax `set user .|<username> session 1|2|3|4|* [auto on|off]
[type off|telnet]`

```
set user .|<username> session 1|2|3|4|* telnet-options  
[host <config_host>] [port <TCP_port>]  
[termtype <terminal_name>] [line-mode on|off]  
[map-cr-crlf on|off] [local-echo on|off]  
[echo <00-7f>] [eof <00-7f>] [erase <00-7f>] [intr <00-7f>]  
[quit <00-7f>]
```

Show Default User

Description Shows the Default User's settings.

User Level Admin

Syntax `show default user`

Show User

Description Shows user configuration settings.

User Level Admin

Syntax `show user <configured_user>|.`

Options `<configured_user>`
Show the settings for the specified user.
.
Show the settings for the current user.

Line Commands

Line Commands

Set Line

Description Configures line parameters.

User Level Normal, Admin

Syntax `set line [data-bits 5|6|7|8] [dial none|in|out|both]
[idle-timer <0-4294967>] [line-name <name>]
[modem-name <config_modem>] [pages 1|2|3|4|5|6|7]
[parity none|even|odd|mark|space] [phone-number <phone_number>]
[rev-sess-security on|off] [sess-timer <0-4294967>]
[stop-bits 1|2|1.5] [termtype wyse60|vt100|ansi|dumb|tvi925|
ibm3151te|vt320|hp700|term1|term2|term3]`

Admin `set line ... [break on|off] [flowin on|off] [flowout on|off]
[hotkey-prefix <00-7f>] [initial cli|menu] [keepalive on|off]
[lock on|off] [mode enabled|disabled] [motd on|off]
[reset on|off] [dial-timeout <number>] [dial-retries <number>]
[single-character on|off] [user <name>] [nouser]
[internet-address <IPv4_address>]`

Options **data-bits**

Specifies the number of bits in a byte. The default is **8**.

dial

Determines how a modem will work on the line. If your user is remote and will be dialing in via modem or ISDN TA, set this parameter to **In**; if the Device Server is being used as a router, set this parameter to either **In**, **Out**, or **Both**, depending on which end of the link your Device Server is situated and how you want to initiate the communication.

idle-timer

Enter a time period, in seconds, for which the **Idle Timer** will run. Use this timer to close a connection because of inactivity. When the **Idle Timer** expires, the Device Server will end the connection. The maximum value is 4294967 seconds (about 49 days). The default value of **0** (zero) means the **Idle Timer** will not expire, so the connection is permanently open.

line-name

Provide a name for the line so it can be easily identified.

modem-name

The name of the predefined modem that is used on this line.

pages

For **DSLogin** line service, this is the number of video pages the terminal supports. Valid values are 1-7. The default is **5** pages.

parity

Specifies if you are using **Even**, **Odd**, or **No parity** on the line. If you want to force a parity type, you can specify **Mark** for 1 or **Space** for 0.

phone-number

The phone number to use when **Dial** is set to **Out**.

rev-sess-security

Enables/disables login/password authentication, locally or externally, on reverse Telnet connections. The default is **Off**.

sess-time

Enter a time, in seconds, for which the **Session Timer** will run. Use this timer to forcibly close the session (connection). When the **Session Timer** expires, the Device Server will end the connection. The default value is **0** seconds so the port will never timeout. The maximum value is 4294967 seconds (about 49 days).

break

Specifies how a break is interpreted:

- **off**—The Device Server ignores the break key completely and it is not passed through to the host. This is the default setting.
- **local**—The Device Server deals with the break locally. If the user is in a session, the break key has the same effect as a hot key.
- **remote**—When the break key is pressed, the Device Server translates this into a telnet break signal which it sends to the host machine.
- **break-interrupt**—On some systems such as SunOS, XENIX, and AIX, a break received from the peripheral is not passed to the client properly. If the client wishes to make the break act like an interrupt key (for example, when the stty options **-ignbrk** and **brkintr** are set).

flowin

Determines if input flow control is to be used. Default is **On**. This is active only when **Line Flow Control** is set to **Soft**, **Hard**, or **Both**.

flowout

Determines if output flow control is to be used. Default is **On**. This is active only when **Line Flow Control** is set to **Soft**, **Hard**, or **Both**.

hotkey-prefix

The prefix that a user types to lock a line or redraw the Menu. The default value is **hex 01**, which corresponds to **Ctrl-a (^a)** (hex value 02 would be Ctrl-b (^b), etc.):

- **^a l**—(Lowercase L) Locks the line until the user unlocks it. The user is prompted for a password (any password, excluding spaces) and locks the line. Next, the user must retype the password to unlock the line.
- **^r**—When you switch from a session back to the Menu, the screen may not be redrawn correctly. If this happens, use this command to redraw it properly.

You can use the **Hotkey Prefix** key to lock a line only when the **Line Lock** parameter is **On**.

initial

Specifies the initial interface a user navigates when logging into the line; either the **Menu** or a prompt for the **CLI**. The default is **CLI**.

keepalive

Enables a per-connection TCP keepalive feature; after approximately 3 minutes of network connection idle time, the connection will send a gratuitous ACK to the network peer, either ensuring the connection stays active OR causing a dropped connection condition to be recognised by the reverse raw service.

Applications using this feature need to be aware that there might be some considerable delay between a network disconnection and the port being available for the next connection attempt; this is to allow any data sent on prior connections to be transmitted out of the serial port buffer. Application network retry logic needs to accommodate this feature.

lock

When enabled, the user can lock his terminal with a password using the **Hotkey Prefix** (default Ctrl-a) **^a l** (lowercase L). The Device Server prompts the user for a password and a confirmation.

mode

Enables/disables the line (available only on the TS2 model). The default is enabled.

motd

Enables/disables the message of the day on the line.

user

For **DSLogin** line service, makes this a line that is dedicated to the specified user. Only this user will be able to log in on this line and they won't need to enter their login name - just their password.

nouser

Blanks out the User parameter, in case you want to change a dedicated user line to an undedicated line.

reset

Resets the terminal type connected to the line when a user logs out.

dial-timeout

The number of seconds the Device Server will wait to establish a connection to a remote modem. The default value is **45** seconds.

dial-retries

The number of times the Device Server will attempt to establish a connection with a remote modem. The default value is **2**.

single-character

When enabled, causes the Device Server to process every character as it comes in, as opposed to buffering the characters before processing; this provides better latency at the expense of efficiency.

stop-bits

Specifies the number of stop bits that follow a byte.

term-type

Specifies the type of terminal connected to the line:

- **Dumb**
- **WYSE60**
- **VT100**
- **ANSI**
- **TVI925**
- **IBM3151TE**
- **VT320** (specifically supporting VT320-7)
- **HP700** (specifically supporting HP700/44)
- **Term1, Term2, Term3** (user defined terminals)

internet-address

Used with reverse sessions, users can access serial devices connected to the Device Server by the specified Internet Address (or host name that can be resolved to the Internet Address in a DNS network). You must reboot the Device Server for the **Internet Address** to take affect (the kill line option does not apply to this parameter).

Set Line Interface

Description Configures line interface (hardware) parameters.

User Level Admin

Syntax

```
set line interface eia-232 [monitor-dcd on|off]
[monitor-dsr on|off] [flow none|soft|hard|both]
[speed 50|75|110|134|150|200|300|600|1200|1800|2400|4800|9600|
19200|38400|57600|115200|230400|28800|custom <baud_rate>]

set line interface eia-422 [flow none|soft|hard|both]
[speed 50|75|110|134|150|200|300|600|1200|1800|2400|4800|
9600|19200|38400|57600|115200|230400|28800|custom <baud_rate>]

set line interface eia-485 [tx-driver-control auto|rts]
[flow none|soft]
[duplex full|duplex half [echo-suppression on|off]]
[speed 50|75|110|134|150|200|300|600|1200|1800|2400|4800|
9600|19200|38400|57600|115200|230400|28800|custom <baud_rate>]
```

Options eia-232 | eia422 | 485

Specifies the type of line that is being used with the Device Server. Select either **EIA-232**, **EIA-422**, or **EIA-485** (TS2 supports only EIA-232).

monitor-dcd

Specifies whether the RS-232 signal DCD (Data Carrier Detect) should be monitored. This is used with modems or any other device that sends a DCD signal. When it is monitored and the Device Server detects a DCD signal, the line service is started. Default is **Off**. If both **Monitor DCD** and **Monitor DSR** are enabled, both signals must be detected before the line service is started.

monitor-dsr

Specifies whether the RS-232 signal DSR (data set ready) should be monitored. This is used with modems or any device that sends a DSR signal. When it is monitored and the Device Server detects a DSR signal, the line service is started. Default is **Off**. If both **Monitor DCD** and **Monitor DSR** are enabled, both signals must be detected before the line service is started.

flow

Defines whether the data flow is handled by the software (**Soft**), hardware (**Hard**), **Both**, or **None**.

tx-driver-control

Used with a **EIA-485** serial interface, if your application supports **RTS** (Request To Send), select this option. Otherwise, select **Auto**. Default is **Auto**.

duplex

Specify whether the line is **Full Duplex** (communication both ways at the same time) or **Half Duplex** (communication in one direction at a time).

echo-suppression

This parameter applies only to **EIA-485 Half Duplex** mode. All characters will be echoed to the user and transmitted across the serial ports. Some EIA-485 applications require local echo to be enabled in order to monitor the loopback data to determine that line contention has occurred. If your application cannot handle loopback data, echo suppression should be **On**. The default is echo suppression **Off**.

speed

Specifies the baud rate of the line; keep in mind that speed is affected by the length of the cable. You can also specify a custom baud rate; valid values are 50-230400.

Set Line Service

Description Sets the service for the line. For services that need further configuration, see [Line Service Commands](#) on page 168 to find the Line Service that you want to configure.

User Level Admin

Syntax `set line .|<number>|* service bidir <config_host> <server_port> <host_port>`

`set line .|<number>|* service direct|silent raw <config_host> <host_port>`

`set line .|<number>|* service direct|silent telnet <config_host> [<host_port>]`

`set line .|<number>|* service reverse raw|telnet <server_port>`

`set line .|<number>|* service client-tunnel <config_host> <host_port>`

`set line .|<number>|* service server-tunnel <server_port>`

`set line .|<number>|* service dslogin|udp|vmodem|modbus-master|modbus-slave|custom-app`

`set line .|<number>|* service trueport client-initiated off <config_host> <host_port>`

`set line .|<number>|* service trueport client-initiated on <server_port>`

Options **bidir**

Allows a bidirectional connection on a port.

<config_host>

The name of the target host.

<server_port>

The Device Server port number.

<host_port>

The port number the target host is listening on for incoming connections.

direct

Direct connections bypass the Device Server, enabling the user to log straight into a specific host. A direct connection is recommended where a user logging in to the Device Server is not required. It is also recommended where multiple sessions are not a requirement. The message **Press return to continue** is displayed on the user's screen. The user must press a key to display the host login prompt. The message is redisplayed on logout.

silent

Silent connections are the same as direct connections, except they are permanently established. The host login prompt is displayed on the screen. Logging out redisplay this prompt. Silent connections, unlike direct connections, however, make permanent use of pseudo tty resources and therefore consume host resources even when not in use.

raw

Creates a connection where no authentication takes place and data is passed unchanged.

telnet

Sets the line for a telnet connection.

reverse

Enables a TCP/IP host to establish a login connection on an external machine attached to a port. For example, to access machines like protocol converters, statistical multiplexors, or machines like routers, firewalls, servers, etc.

dslogin

The default connection. The Device Server displays a login on that line. For example, **DSLogin** is used when a System Administrator configures the Device Server or users starts a session(s) from the Device Server to hosts.

udp

Sets the line to listen for and/or send UDP data.

vmodem

The Device Server port behaves as if it were a modem to the attached device.

client-tunnel

Sets the line for a client tunnel connection.

server-tunnel

Sets the line for a server tunnel connection.

modbus-slave

Sets the line to act as a Modbus master.

modbus-master

Sets the line to act as a Modbus slave.

custom-app

Sets the line to use the custom application created with the SDK.

trueport

Sets the line to communicate with the TruePort utility. You must install the TruePort utility on the host machine.

client-initiated

When this option is turned on, the Device Server will wait for a connection from the TruePort host (see the TruePort documentation for information on how to set up this feature on the TruePort host). When this option is turned off, the Device Server will initiate the connection to the TruePort host. The default is off.

Set Modem

Description Sets the modem initialization strings.

User Level Admin

Syntax **set modem** *<modem_name>* *<init_string>*

Options *<modem_name>*

Predefined modem name.

<init_string>

Specify the initialization string for the internal modem. This can be up to 60 characters long, but cannot include spaces.

Set Termtyp

Description Sets the terminal type for the current terminal session. term1, term2, and term3 refer to the user-uploadable custom terminal definitions. If these are not present, the default is wyse60.

User Level Restricted, Normal, Admin

Syntax **set termtyp**
[wyse60|vt100|ansi|dumb|tvi925|ibm3151te|vt320|hp700|term1|term2|term3]

Option wyse60|vt100|ansi|dumb|tvi925|ibm3151te|vt320|hp700|term1|term2|term3

Specifies the type of terminal connected to the line:

- **Dumb**
- **WYSE60**
- **VT100**
- **ANSI**
- **TVI925**
- **IBM3151TE**
- **VT320** (specifically supporting VT320-7)
- **HP700** (specifically supporting HP700/44)
- **Term1, Term2, Term3** (user defined terminals)

Show Line

Description Shows the line settings/information.

User Level Admin

Syntax **show line**

Line Service Commands

Set Custom-App

Description You can create a custom application that can run on a specific serial line in Device Server using the Perle SDK.

User Level Admin

Syntax **set custom-app line** .|<number>|* **program-command-line** <command>

Options **program-command-line**

The name of the SDK program executable that has been already been downloaded to the Device Server, plus any parameters you want to pass to the program. Maximum of 80 characters. Use the **shell** CLI command as described in the *SDK Programmer's Guide* to manage the files that you have downloaded to the Device Server. For example, using sample outraw program, you would type:

outraw 192.168.2.1:10001 Acct:10001

if you were starting the application on a line.

Set Telnet-Client

Description Configures telnet parameters.

User Level Normal, Admin

Syntax **set telnet-client** [termttype <terminal_name>] [line-mode on|off] [map-cr-crlf on|off] [local-echo on|off] [echo <00-7f>] [eof <00-7f>] [erase <00-7f>] [intr <00-7f>] [quit <00-7f>] [escape <00-7f>]

Options **termttype**

Type of terminal attached to this line; for example, ANSI or WYSE60.

line-mode

When **On**, keyboard input is not sent to the remote host until **Enter** is pressed, otherwise input is sent every time a key is pressed. Default is **Off**.

map-cr-crlf

Maps carriage returns (CR) to carriage return line feed (CRLF). The default value is **Off**.

local-echo

Toggles between local echo of entered characters and suppressing local echo. Local echo is used for normal processing, while suppressing the echo is convenient for entering text that should not be displayed on the screen, such as passwords. This parameter can only be used when **Line Mode** is **On**. Default is **Off**.

echo

Defines the echo character. When Line Mode is On, typing the echo character echoes the text locally and sends only completed lines to the host. This value is in hexadecimal with a default value of **5** (ASCII value **^E**).

eof

Defines the end-of-file character. When Line Mode is On, entering the eof character as the first character on a line sends the character to the remote host. This value is in hexadecimal with a default value of **4** (ASCII value **^D**).

erase

Defines the erase character. When Line Mode is Off, typing the erase character erases one character. This value is in hexadecimal with a default value of **8** (ASCII value **^H**).

intr

Defines the interrupt character. Typing the interrupt character interrupts the current process. This value is in hexadecimal with a default value of **3** (ASCII value **^C**).

quit

Defines the quit character. Typing the quit character closes and exits the current telnet session. This value is in hexadecimal with a default value of **1c** (ASCII value **FS**).

escape

Defines the escape character. Returns you to the command line mode. This value is in hexadecimal with a default value of **1d** (ASCII value **GS**).

Set UDP

Description	Configures the UDP settings for the serial line.
User Level	Normal, Admin
Syntax	set udp line entry 1 2 3 4 [both in out none <outbound_port>] [<start_ip_address>] [<end_ip_address>]
Options	both in out none The direction in which information is received or relayed: • None—UDP service not enabled. • In—LAN to serial. • Out—Serial to LAN. • Both—Messages are relayed both directions. <outbound_port> The port that the Device Server will use to relay messages to servers/hosts. The Device Server will listen for UDP packets on the port configured by the DS Port parameter. <start_ip_address> The first host IP address in the range of IP addresses (for IPV4 or IPV6) that the Device Server will listen for messages from and/or send messages to. <end_ip_address> The last host IP address in the range of IP addresses (for IPV4, not required for IPV6) that the Device Server will listen for messages from and/or send messages to.

Set Vmodem

Description	Configures the vmodem settings for the serial line.
User Level	Admin
Syntax	set vmodem line [failure-string <string>] [success-string <string>] [host <config_host>] [port <TCP_port> 0] [response-delay <time_ms>] [style numeric verbose] [suppress on off]
Options	failure-string String that is sent to the serial device when a connection fails. If no string is entered, then the string NO CARRIER will be sent. success-string String that is sent to the serial device when a connection succeeds. If no string is entered, then the string CONNECT will be sent with the connecting speed, for example CONNECT 9600.

host

The target host name.

port

The amount of time, in milliseconds, before an AT response is sent to the requesting device. The default is 250 ms.

response-delay

The port number the target host is listening on for messages.

style

One of the following:

- **Verbose**—Return codes (strings) are sent to the connected device.
- **Numeric**—The following characters can be sent to the connected device:
 - 1 Successfully Connected
 - 2 Failed to Connect
 - 4 Error

suppress

If set to **No**, connection success/failure indication strings are sent to the connected device, otherwise these indications are suppressed.

Set Modbus-Slave Line

Description Sets the Modbus slave parameters for the line. SSL/TLS can be enabled and configured for this Line Service.

User Level Admin

Syntax `set modbus-slave line .|<number>|* [crlf on|off]
[protocol rtu|ascii] [uid-range <uid_range>]`

Options **crlf**

When **Modbus/ASCII** is selected, adds a CR/LF to the end of the transmission; most Modbus devices require this option. The default is **On**.

protocol

Specify the protocol that is used between the Modbus Master(s) and Modbus Slave(s), either RTU or ASCII.

uid-range

You can specify a range of UIDs (1-247), in addition to individual UIDs. The format is comma delimited; for example, 2-35, 50, 100-103.

Set Modbus-Master Line

Description Sets the Modbus master parameters for the line. SSL/TLS can be enabled and configured for this Line Service.

User Level Admin

Syntax `set modbus-master line .|<number>|* [crlf on|off]
[protocol rtu|ascii]
[[entry <number> [port <port>] [protocol udp|tcp]
[range-mode gateway|host] [slave-ip <IP_address>]
[uid-range <start_uid> <end_uid>]]`

Options **crlf**

When **Modbus/ASCII** is selected, adds a CR/LF to the end of the transmission; most Modbus devices require this option. The default is **On**.

protocol

Specify the protocol that is used between the Modbus Master(s) and Modbus Slave(s), either RTU or ASCII.

entry

You can specify up to 16 Modbus Slave Remote IP Mapping entries (the UIDs must not overlap).

port

The destination port of the remote Modbus TCP Slave that the Device Server will connect to.

protocol

Specify the protocol that is used between the Modbus Master and Modbus Slave(s), either TCP or UDP.

range-mode

If you specify **Host**, the IP address is used for the first UID specified in the range. The last octet in the IPv4 address is then incremented for subsequent UID's in that range. The **Host** option is not applicable for IPv6 addresses. If you specify **Gateway**, the Modbus Master Gateway will use the same IP address when connecting to all the remote Modbus slaves in the specified UID range.

slave-ip

The IP address of the TCP/Ethernet Modbus Slave.

uid-range

When **Range Mode** is **Host** and you have sequential Modbus Slave IP addresses (for example, 10.10.10.1, 10.10.10.2, 10.10.10.3, etc.), you can specify a UID range and the Device Server will automatically increment the last digit of the configured IP address. Therefore, you can specify a UID range of 1-100, and the Device Server will route Master Modbus messages to all Modbus Slaves with IP addresses of 10.10.10.1 - 10.10.10.100.

Show Custom-App

Description Shows the custom application line settings.

User Level Admin

Syntax `show custom-app line .|<number>|*`

Show Interface

Description Shows the network interface information.

User Level Admin

Syntax **show interface** [**brief|ethernet**]

Show Modbus

Description Shows the Modbus settings for a line.

User Level Admin

Syntax **show modbus master|slave** <number>

Show Telnet-Client

Description Shows the telnet client settings for a line.

User Level Admin

Syntax **show telnet-client**

Show UDP

Description Shows the UDP settings for the line.

User Level Admin

Syntax **show udp**

Show Vmodem

Description Show the vmodem settings for the line.

User Level Normal, Admin

Syntax **show vmodem**

Modem Commands

Add Modem

Description Adds a modem.

User Level Admin

Syntax **add modem** <modem_name> <initialization_string>

Options <modem_name>

The name of the modem. Do not use spaces.

<initialization_string>

The initialisation string of the modem; see your modem's documentation.

Delete Modem

Description Deletes a modem.

User Level Admin

Syntax **delete modem** <config_modem_name>

Option <config_modem_name>

You can see a the list of modems that can be deleted by typing **delete modem ?**.

Show Modems

Description Shows the Device Server modem table.

User Level Normal, Admin

Syntax **show modems**

Packet Forwarding Commands

Set Packet-Forwarding Line

Description The Packet Forwarding feature allows you to control how the data coming from a serial device is packetized before forwarding the packet onto the LAN network. This command configures packet forwarding options for serial devices attached to the serial line. The command is broken up into the two logical flows that can be configured; if you configure both the packet options and the frame definition options, the frame definition options will take precedence. If any of the packet options that are configured are met, the packet transmission is triggered.

User Level Admin

Syntax

```
set packet-forwarding line <number>|*
[enable-end-trigger1 on|off] [enable-end-trigger2 on|off]
[end-trigger1 <0x0-FF>] [end-trigger2 <0x0-FF>]
[force-transmit-timer <number>]
[forwarding-rule trigger1|trigger+1|trigger+2|strip-trigger]
[idle-timer <number>] [packet-size <number>]

set packet-forwarding line <number>|*
[enable-eof1 on|off] [enable-eof2 on|off] [enable-sof1 on|off]
[enable-sof2 on|off] [eof1 <0x0-FF>] [eof2 <0x0-FF>]
[forwarding-rule trigger1|trigger+1|trigger+2|strip-trigger]
[sof1 <0x0-FF>] [sof2 <0x0-FF>] [start-frame-transmit on|off]
```

Options **enable-end-trigger1**

Enable or disable the end trigger1 hex character.

enable-end-trigger2

Enable or disable the end trigger2 hex character.

enable-end-eof1

Enable or disable the eof1 (end of frame) hex character.

enable-end-eof2

Enable or disable the eof2 (end of frame) hex character.

enable-end-sof1

Enable or disable the sof1 (start of frame) hex character.

enable-end-sof2

Enable or disable the sof2 (start of frame) hex character.

end-trigger1

When enabled, specifies the character that when received will define when the packet is ready for transmission. The actual transmission of the packet is based on the Trigger Forwarding Rule. Valid values are in hex 0-FF. The default is 0.

end-trigger2

When enabled, creates a sequence of characters that must be received to specify when the packet is ready for transmission (if the End Trigger1 character is not immediately followed by the End Trigger2 character, the Device Server waits for another End Trigger1 character to start the End Trigger1/End Trigger2 character sequence). The actual transmission of the packet is based on the Trigger Forwarding Rule. Valid values are in hex 0-FF. The default is 0.

eof1

Specifies the End of Frame character, which defines when the frame is ready to be transmitted. The actual transmission of the frame is based on the Trigger Forwarding Rule. Valid values are in hex 0-FF. The default is 0.

eof2

When enabled, creates a sequence of characters that must be received to define the end of the frame (if the EOF1 character is not immediately followed by the EOF2 character, the Device Server waits for another EOF1 character to start the EOF1/EOF2 character sequence), which defines when the frame is ready to be transmitted. The actual transmission of the frame is based on the Trigger Forwarding Rule. Valid values are in hex 0-FF. The default is 0.

force-transmit-timer

When the specified amount of time, in milliseconds, elapses after the first character is received from the serial port sender, the packet is transmitted. A value of zero (0) ignores this parameter. Valid values are 0-65535 ms. The default is 0.

forwarding-rule

Determines what is included in the Frame (based on the EOF1 or EOF1/EOF2) or Packet (based on Trigger1 or Trigger1/Trigger2). Choose one of the following options:

- **Strip-Trigger**—Strips out the EOF1, EOF1/EOF2, Trigger1, or Trigger1/Trigger2, depending on your settings.
- **Trigger**—Includes the EOF1, EOF1/EOF2, Trigger1, or Trigger1/Trigger2, depending on your settings.
- **Trigger+1**—Includes the EOF1, EOF1/EOF2, Trigger1, or Trigger1/Trigger2, depending on your settings, plus the first byte that follows the trigger.
- **Trigger+2**—Includes the EOF1, EOF1/EOF2, Trigger1, or Trigger1/Trigger2, depending on your settings, plus the next two bytes received after the trigger.

idle-timer

The amount of time, in milliseconds, that must elapse between characters before the packet is transmitted to the network. A value of zero (0) ignores this parameter. Valid values are 0-65535 ms. The default is 0.

packet-size

The number of byte that must be received from the serial port before the packet is transmitted to the network. A value of zero (0) ignores this parameter. Valid values are 0-1024 bytes. The default is 0.

sof1

When enabled, the Start of Frame character defines the first character of the frame, any character(s) received before the Start of Frame character is ignored. Valid values are in hex 0-FF. The default is 0.

sof2

When enabled, creates a sequence of characters that must be received to create the start of the frame (if the SOF1 character is not immediately followed by the SOF2 character, the Device Server waits for another SOF1 character to start the SOF1/SOF2 character sequence). Valid values are in hex 0-FF. The default is 0.

start-frame-transmit

When enabled, the SOF1 or SOF1/SOF2 characters will be transmitted with the frame. If not enabled, the SOF1 or SOF1/SOF2 characters will be stripped from the transmission.

Show Packet-Forwarding Line

Description Shows the packet-forwarding settings for the line.
User Level Admin
Syntax **show packet-forwarding line** <number>

Network Commands

SNMP Commands

The Device Server supports SNMP traps restart and SNMP community authentication error.

Add Community

Description Adds an SNMP community (version 1 and version 2).
User Level Admin
Syntax **add community** <community_name> <config_host>|<ip_address>
none|readonly|readwrite
Options <community_name>
 A name that will be sent to the Device Server from an SNMP manager. This name will define the permissions of the manager.
 <config_host>|<ip_address>
 The host name of the SNMP community that will send requests to the Device Server.
 The IP address of the SNMP manager that will send requests to the Device Server. If the address is 0.0.0.0, any SNMP manager with the **Community Name** can access the Device Server.
none|readonly|readwrite
 Permits the Device Server to respond to SNMP requests by:

- **None**—There is no response to requests from SNMP.
- **Readonly**—Responds only to Read requests from SNMP.
- **Readwrite**—Responds to both Read and Write requests from SNMP.

Add Trap

Description Adds an SNMP trap.
User Level Admin
Syntax **add trap** <trap_name> <config_host>|<ip_address>
Options <trap_name>
 An arbitrary trap community name.
 <config_host>|<ip_address>
 Defines the hosts (by IP address) that will receive trap messages generated by the Device Server. Up to four trap hosts can be defined.

Delete Community

Description Deletes an SNMP community (version 1 and version 2).

User Level Admin

Syntax **delete community** *<config_community_number>*

Option *<config_community_number>*

When you add an SNMP community, it gets assigned to a number. To delete the SNMP community, you need to specify the number of the community that you want to delete. To see which community is assigned to what number, type the **show snmp** command.

Delete Trap

Description Deletes an SNMP trap.

User Level Admin

Syntax **delete trap** *<config_trap_number>*

Option *<config_trap_number>*

When you add an SNMP trap, it gets assigned to a number. To delete the SNMP trap, you need to specify the number of the trap that you want to delete. To see which trap is assigned to what number, type the **show snmp** command.

Set SNMP

Description Configures SNMP settings.

User Level Admin

Syntax **set snmp** [**contact** *<string>*] [**location** *<string>*]
[**readonly user** *<username>*] [**readwrite user** *<username>*]

Options **contact**

The name and contract information of the person who manages this SMNP node.

location

The physical location of the SNMP node.

readonly user

(SNMP version 3) Specified user can only view SNMP variables.

readwrite user

(SNMP version 3) Specified user can view and edit SNMP variables.

Show SNMP

Description Shows SNMP settings, including communities and traps.

User Level Admin

Syntax **show snmp**

TFTP Commands

Set Server TFTP

Description Configures the Device Server's TFTP client settings.

User Level Admin

Syntax **set server tftp** [**retry** *<integer>*] [**timeout** *<integer>*]

Options **retry**

The number of times the Device Server will attempt to transfer (using TFTP) a file to/from a host. Enter a value between 0 and 5. The default is **5**. A value of **0** (zero) means that the Device Server will not attempt a retry should TFTP fail.

timeout

The time, in seconds, that the Device Server will wait for a successful transmit or receipt of TFTP packets before retrying a TFTP transfer. Enter a value between 3 and 10. The default is **3** seconds.

Hosts Commands

Add Host

Description Adds a host to the Device Server host table.

User Level Admin

Syntax **add host** *<hostname>* *<ip_address>*

Options *<hostname>*

The name of the host.

<ip_address>

The host IP address.

Delete Host

Description Deletes a host from the Device Server host table.

User Level Admin

Syntax **delete host** *<config_host>*

Option *<config_host>*

You can see a list of hosts that can be deleted by typing **delete host ?**.

Set Host

Description Configures a host in the Device Server host table.

User Level Admin

Syntax **set host** *<config_host>* *<ip_address>*

Options *<config_host>*

The name of the host.

<ip_address>

The host IP address.

Show Hosts

Description Shows the Device Server host table.

User Level Normal, Admin

Syntax **show hosts**

Gateway Commands

Add Gateway

Description Adds a gateway. You can configure up to twenty gateways.

User Level Admin

Syntax **add gateway** <config_host> **default**

add gateway <config_host> **host** <dest_IP_addr>

add gateway <config_host> **network**
<dest_IPv4_addr>|<dest_IPv6_addr>
[<subnet_bits_0-32>|<subnet_bits_0-128>]

Options <config_host>

You can specify up to twenty hosts to act as gateways in your network. Each gateway host must be defined in the Device Server host table.

default|host|network

Specify the type of gateway:

- **Default**—A gateway which provides general access beyond your local network.
- **Host**—A gateway reserved for accessing a specific host external to your local network.
- **Network**—A gateway reserved for accessing a specific network external to your local network.

<dest_IP_addr>

When the gateway is a **Host** or **Network** gateway, you must specify the IP address of the target host machine/network.

<subnet_bits>

When the gateway is a **Network** gateway, you must specify the network's subnet mask.

Delete Gateway

Description Deletes a gateway.

User Level Admin

Syntax **delete gateway** <config_gateway_host>

Option <config_gateway_host>

You can view the configured gateways that can be deleted by typing **delete gateway ?**.

Set Gateway

Description Configures the gateway.

User Level Admin

Syntax **set gateway** *<config_gateway_host>* **default**

set gateway *<config_gateway_host>* **host** *<destination_ip>*

set gateway *<config_gateway_host>*

network *<dest_IPv4_addr>|<dest_IPv6_address> <prefixbits_mask>|*

Options *<config_gateway_host>*

You can view the configured gateways that can be deleted by typing **delete gateway ?**.

default|host|network

Specify the type of gateway:

- **Default**—A gateway which provides general access beyond your local network.
- **Host**—A gateway reserved for accessing a specific host external to your local network.
- **Network**—A gateway reserved for accessing a specific network external to your local network.

<destination_ip>

When the gateway is a **Host** or **Network** gateway, you must specify the IP address of the target host machine/network.

<prefixbits_mask>

When the gateway is a **Network** gateway, you must specify the network's subnet mask for an IPv4 destination IP address (the address is in the form of 123.123.123.123) or prefix bits for an IPv6 destination IP address (valid values are 0-128).

Show Gateways

Description Shows configured gateways.

User Level Normal, Admin

Syntax **show gateways**

Logging Commands

Set Syslog

Description Configures the system log.

User Level Admin

Syntax **set syslog**
[**level** **emergency|alert|critical|error|warning|notice|info|debug**]
[**primary-host** <*config_host*>] [**secondary-host** <*config_host*>]

Options **level**

Choose the event level that triggers a syslog entry:

- **Emergency**
- **Alert**
- **Critical**
- **Error**
- **Warning**
- **Notice**
- **Info**
- **Debug**

When you select a **Level**, all the levels that appear above it in the list also trigger a syslog entry. For example, if you select **Error**, all **Error**, **Critical**, **Alert**, and **Emergency** events will be logged.

primary-host

The first preconfigured host that the Device Server will attempt to send system log messages to; messages will be displayed on the host's monitor.

secondary-host

If the Device Server cannot communicate with the primary host, then the Device Server will attempt to send system log messages to this preconfigured host; messages will be displayed on the host's monitor.

Show Syslog

Description Shows the syslog settings.

User Level Admin

Syntax **show syslog**

Time Commands

Set Time

Description Sets the Device Server's system clock.

User Level Admin

Syntax **set time** <*hh:mm[:ss]*>

Option <*hh:mm[:ss]*>

Sets the Device Server's system time, using military time format.

Show Time

Description Shows the Device Server's system clock.

User Level Normal, Admin

Syntax **show time**

Time/Date Setting Commands

Set Date

Description Sets the Device Server's system clock.

User Level Admin

Syntax `set date <dd/mm/yyyy>`

Set Time

Description Sets the Device Server's system clock.

User Level Admin

Syntax `set time <hh:mm[:ss]>`

Option `<hh:mm[:ss]>`

Sets the Device Server's system time, using military time format.

Show Date

Description Shows the date, according to the Device Server system clock.

User Level Normal, Admin

Syntax `show date`

Show Time

Description Shows the Device Server's system clock.

User Level Normal, Admin

Syntax `show time`

Administration Commands

Bootup Commands

Reboot

Description Reboots the Device Server. You will be prompted to save configuration to FLASH, if there have been unsaved configuration changes.

User Level Admin

Syntax `reboot`

Reset

Description Resets the user profile or serial line to the default factory configuration.

User Level Admin

Syntax `reset user .|<username>|*`

`reset line`

Reset Factory

Description Resets the Device Server to the factory configuration.

User Level Admin

Syntax `reset factory`

Save

Description Saves the configuration to FLASH.

User Level Admin

Syntax `save`

Set Bootup

Description Specifies remote the TFTP host and pathname for files to be loaded after a Device Server reboot.

User Level Admin

Syntax **set bootup firmware host** <hostname> [**file** <path_filename>]

set bootup configuration host <hostname> [**file** <path_filename>]

Options **firmware file**

The path and file name (do not use a drive letter), relative to the default path of your TFTP server software, of the update software for the Device Server that will be loaded when the Device Server is rebooted.

configuration file

The path and file name (do not use a drive letter), relative to the default path of your TFTP server software, of the configuration software for the Device Server that will be loaded when the Device Server is rebooted.

host

The host name or IP address of the server that contains the configuration or firmware file. If you use a host name, it must exist in the Device Server's host table or be resolved by DNS.

Show ARP

Description Shows the current contents of the ARP cache.

User Level Admin

Syntax **show arp**

Show Bootup

Description Shows the Firmware and Configuration files specified for Device Server bootup.

User Level Admin

Syntax **show bootup**

TFTP File Transfer Commands

Netload

Description Transfers a file from a remote host to the Device Server using the TFTP protocol.

User Level Admin

Syntax **netload firmware|configuration|customlang|term1|term2|term3**
customapp-file <hostname/ip_address> <filename>

Options **firmware**

Specifies that you are going to download a new firmware file to the Device Server.

configuration

Specifies that you are going to download a new configuration file to the Device Server.

customlang

Specifies that you are going to download a custom language file to the Device Server.

term1|term2|term3

You can create and download up to three custom terminal definitions to the Device Server.

customapp-file

You can download multiple SDK program executables and ancillary files using this command by running the command multiple times to download multiple files. Use the **shell** CLI command as described in the *SDK Programmer's Guide* to manage the files that you download.

<hostname/ip_address>

The IP address or host name where the file you are downloading to the Device Server resides. If you are using a host name, it must be resolved in either the Device Server's **Host Table** or a DNS server.

<filename>

The complete path and file name (cannot use a drive letter) of the file you are downloading to the Device Server.

Netsave

Description Transfers a file from the Device Server to a remote host using the TFTP protocol.

User Level Admin

Syntax **netsave configuration|crash <hostname/ip_address> <filename>**

Options **configuration**

Specifies that you are going to upload a configuration file from the Device Server to the specified host or IP address.

crash

Specifies that you are going to upload a crash file from the Device Server to the specified host or IP address.

<hostname/ip_address>

The IP address or host name for where the file you are uploading from the Device Server is going. If you are using a host name, it must be resolved in either the Device Server's **Host Table** or a DNS server.

<filename>

The complete path and file name (cannot use a drive letter) for the file you are uploading from the Device Server.

MOTD Commands

Set MOTD

Description Specifies the server/file that contains the message of the day (MOTD) that is displayed when users log into the Device Server.

User Level Normal, Admin

Syntax **set motd host <hostname> file <path_filename>**

Options **host**

The host that the Device Server will be getting the Message of the Day file from.

file

The path and file name (do not use a drive letter), relative to the default path of your TFTP server software, of the file that contains a string that is displayed when a user connects to the Device Server.

Show MOTD

Description Show the Message of the Day (MOTD) settings.

User Level Admin

Syntax `show motd`

Statistic Commands

Configuration Statistics

Show Netstat

Description Shows currently used TCP/UDP sockets/ports.

User Level Admin

Syntax `show netstat [all] [listening] [tcp] [udp] [tcpv6] [udpv6]`

Options `all`

Displays all ports, including server (listening) ports; by default, listening ports are not displayed.

listening

Displays server (listening) ports; by default, listening ports are not displayed.

tcp

Displays TCP port statistics.

udp

Displays UDP port statistics.

tcpv6

Displays TCPv6 port statistics.

udpv6

Displays UDPv6 port statistics.

Show Modbus Statistics

Description Shows the Modbus statistics.

User Level Admin

Syntax `show modbus statistics master-tcp line *|<number>`

`show modbus statistics master-udp line *|<number>`

`show modbus statistics slave-tcp line *|<number>`

`show modbus statistics slave-udp line *|<number>`

Show Netstat Statistics

Description Shows protocol (IP/ICMP/TCP/UDP) counters.

User Level Admin

Syntax `show netstat statistics [ip] [ipv6] [icmp] [icmpv6] [tcp] [udp] [udp6]`

Show Routes

Description Shows current information about IPv4 or IPv6 network routes.

User Level Admin

Syntax `show routes [ipv6]`

Run-Time Statistics

Delete Arp

Description Delete entries from the Device Server's ARP cache. Takes effect immediately; not related to configuration.

User Level Admin

Syntax `delete arp`

Show Arp

Description Shows the current contents of the ARP cache.

User Level Admin

Syntax `show arp`

Show Serial

Description Shows statistics on the serial port.

User Level Admin

Syntax `show serial`

Uptime

Description Displays the elapsed time (in days, hours, minutes, and seconds) since the last reboot/power cycle.

User Level Admin

Syntax `uptime`

I/O Commands

Global I/O Commands

Set IO UDP

Description Sets the UDP settings for I/O unicast messages.

User Level Admin

Syntax `set io udp [mode on|off]
[broadcast-interval <broadcast_interval>]`

`set io udp entry 1|2|3|4 disabled`

Options `set io udp entry 1|2|3|4 <udp_port> <start_ip> [<end_ip>]
mode`

Enables/disables UDP broadcast of I/O channel status (data).

broadcast-interval

Enter the interval, in seconds, for UDP broadcasts of I/O channel status (data). Valid values are 1-9999. Default value is 30 seconds.

entry

You can specify up to four sets of UDP IP address that will receive the I/O unicast.

udp_port

The UDP port that the Device Server will use to relay messages to servers/hosts.

start_ip

The first host IP address in the range of IP addresses (for IPV4 or IPV6) that the Device Server will listen for messages from and/or send messages to.

end_ip

The last host IP address in the range of IP addresses (for IPV4, not required for IPV6) that the Device Server will listen for messages from and/or send messages to.

Set IO Failsafe

Description Sets the failsafe (watchdog) settings for I/O.

User Level Admin

Syntax **set io failsafe [mode on|off] [timeout <seconds>]**

Options **mode**

Enables/disables the **Failsafe Timer**. This is the global setting that must be enabled to set the **Failsafe Action** on the channel for digital outputs and relays. When this timer expires because of no I/O activity within the specified time interval, the **Failsafe Action** set for the channel determines the action on the output.

timeout

The number of seconds that must elapse with no I/O activity before the channel **Failsafe Action** is triggered. Valid values are 1-9999. The default is 30 seconds.

Set IO Modbus

Description Enabling the Modbus option makes the Device Server act as a Modbus Slave, allowing Modbus Masters to communicate with the Device Server to control and/or retrieve I/O data.

User Level Admin

Syntax **set io modbus [mode on|off] [uid <1-255>]**

Options **mode**

Enables/disables the ability to control/monitor I/O channels using the Modbus protocol.

uid

This is the UID you are assigning to the Device Server, which is acting as a Modbus slave.

Set IO Temperature-Scale

Description Sets the temperature scale that will be used for all Temperature input readings.

User Level Admin

Syntax **set io temperature-scale celsius|fahrenheit**

Option **temperature-scale**

Select the temperature scale that will be used to display temperature data, either Fahrenheit or Celsius. The default is Celsius.

Set Line

Set Line Service

Description Sets the **Line Service** settings to signal I/O.

User Level Admin

Syntax **set line** *<number>* **service signal-io**

Option **signal-io**

Sets the line to use signal I/O. You still need to define the serial pins for digital input (CTS, DSR, or DCD) or digital output (RTS or DTR). See [Set IOChannel Digital Input \(Serial Pins\)](#) on page 189 or [Set IOChannel Digital Output \(Serial Pins\)](#) on page 191 for configuration options.

Set IOChannel

Set IOChannel Mode

Description Sets general I/O channel settings for the specified channel, these settings are available to all channels and I/O serial pins.

User Level Admin

Syntax **set iochannel** *<i/o_channel>* [**mode enabled|disabled**] [**description** *<string>*]

Options *i/o_channel*

Specify the channel number, for example, d2 or a4. Temperature models use Analog input, so the channel numbers are a1-a4.

mode

Enables the channel, allowing the settings to become active.

description

Provide a description of the channel, making it easier to identify. The channel description can be up to 20 characters.

Set IOChannel Digital I/O

Description Sets up the Digital I/O channel to act as either an output or input channel.

User Level Admin

Syntax **set iochannel** *<digital_channel>* **source-type input|output**

Options *digital_channel*

Specify the Digital channel number, for example, d2.

source-type

Specify whether the channel will drive the line (output) or will be reading the status of the line (input). The default is **Input**. The internal jumpers must match the software configuration, so if you change this setting to **Output**, you will have to also change the internal hardware jumpers.

Set IOChannel Digital Input

Description Sets the Digital input settings for the channel.

User Level Admin

Syntax **set iochannel** <digital_channel>
[alarm [trigger disabled|inactive-input|active-input]
[clear auto|manual] [syslog on|off] [snmp on|off]]
[description <string>] [invert-signal on|off]
[latch disabled|inactive-to-active|active-to-inactive]

Options **digital_channel**

Specify the Digital channel number, for example, d2.

alarm

Configures alarm settings when the Digital input trigger is activated.

trigger

When the trigger condition is met, triggers the specified alarm action. Triggers can be:

- **Disabled**—No alarm settings. This is the default.
- **Inactive**—When the expected Digital input is active, going inactive will trigger an alarm.
- **Active**—When the expected Digital input is inactive, going active will trigger an alarm.

clear

Specify **Manual** to manually clear an alarm. Specify **Auto** to automatically clear the alarm when the trigger condition changes; for example, if the **Trigger** is **Inactive** and the alarm is triggered, once the input becomes active again, the alarm will be cleared when **Auto** is set. The default is **Auto**.

syslog

Sends a message to syslog when an alarm is triggered or cleared. The syslog entry includes the severity level and the value that caused the alarm to trigger or clear. The syslog message is associated with **Level Critical**.

snmp

Sends an SNMP trap when an alarm is triggered or cleared. The trap consists of the severity level and whether the alarm was triggered or cleared.

description

Provide a description of the channel, making it easier to identify. The channel description can be up to 20 characters.

invert-signal

Inverts the actual condition of the I/O signal in the status; therefore, an inactive status will be displayed as active.

latch

Latches (remembers) the activity transition (active-to-inactive or inactive-to-active). The default is disabled.

Set IOChannel Digital Input (Serial Pins)

Description Sets the Digital input settings for serial pins CTS, DSR, and DCD. This option is only available when the **Line Service** is set to **Signal I/O**.

User Level Admin

Syntax **set iochannel cts|dsr|dcd**
[alarm [trigger disabled|inactive-input|active-input]
[clear auto|manual] [syslog on|off] [snmp on|off]]
[description <string>] [invert-signal on|off]
[latch disabled|inactive-to-active|active-to-inactive]

Options **digital_channel**

Specify the Digital channel number, for example, d2.

alarm

Configures alarm settings when the Digital input trigger is activated.

trigger

When the trigger condition is met, triggers the specified alarm action. Triggers can be:

- **Disabled**—No alarm settings. This is the default.
- **Inactive**—When the expected Digital input is active, going inactive will trigger an alarm.
- **Active**—When the expected Digital input is inactive, going active will trigger an alarm.

clear

Specify **Manual** to manually clear an alarm. Specify **Auto** to automatically clear the alarm when the trigger condition changes; for example, if the **Trigger** is **Inactive** and the alarm is triggered, once the input becomes active again, the alarm will be cleared when **Auto** is set. The default is **Auto**.

syslog

Sends a message to syslog when an alarm is triggered or cleared. The syslog entry includes the severity level and the value that caused the alarm to trigger or clear. The syslog message is associated with **Level Critical**.

snmp

Sends an SNMP trap when an alarm is triggered or cleared. The trap consists of the severity level and whether the alarm was triggered or cleared.

description

Provide a description of the channel, making it easier to identify. The channel description can be up to 20 characters.

invert-signal

Inverts the actual condition of the I/O signal in the status; therefore, an inactive status will be displayed as active.

latch

Latches (remembers) the activity transition (active-to-inactive or inactive-to-active). The default is disabled.

Set IOChannel Digital Output

Description Sets the Digital output channel settings.

User Level Admin

Syntax

```
set iochannel <digital_channel>
[type sink|source|sink-and-source] [active-signal-width <width>]
[inactive-signal-width <width>]
[failsafe-action none|activate-output|deactivate-output]

set iochannel <digital_channel>
output [pulse continuous|counted <pulse_count>]
[active-to-inactive-delay <delay>]
[inactive-to-active-delay <delay>]
```

Options *digital_channel*

Specify the Digital channel number, for example, d2.

type

Specify the type of digital output:

- **Sink**—Specifies that the channel will be grounded when active.
- **Source**—Specifies that the channel will provide voltage when active.
- **Sink and Source**—Specifies that channel will be grounded when it is inactive and will provide voltage when it is active.

The default is **Sink**.

active-signal-width

How long the channel output will be active during the pulse mode. Valid values are 1-9999 x 100 ms. The default is 100 ms.

inactive-signal-width

How long the channel output will remain inactive during pulse mode. Valid values are 1-9999 x 100 ms. The default is 100 ms.

failsafe-action

When there has been no I/O activity within the specified time (set in the Global Settings) and the **Failsafe Timer** is triggered, you can set the **Failsafe Action** to:

- **None**—The state of the Digital/Relay output remains the same, no change.
- **Activate Output**—Activates the output.
- **Deactivate Output**—Deactivates the output.

output

Specify how the channel Digital output will be handled:

- **Manual**—You must manually activate and deactivate the output.
- **Pulse**—When the output is activated, this configuration will cause the output to pulse (go active and inactive) at a configured rate.
- **Inactive-to-Active Delay**—The channel output will remain inactive for the specified time interval after it is manually started.
- **Active-to-Inactive Delay**—The channel output will go inactive after the specified time interval after it is manually started.

The default is **Manual**.

pulse

When the **Output** is **Pulse**, you can have it pulse in a **Continuous** manner or specify a pulse **Count** (each count consists of an active/inactive sequence). The default is **Continuous**.

active-to-inactive-delay

How long to delay an active-to-inactive or inactive-to-active setting after it is manually started. Valid values are 1-9999 x 100 ms. The default is 100 ms.

inactive-to-active-delay

How long to delay an active-to-inactive or inactive-to-active setting after it is manually started. Valid values are 1-9999 x 100 ms. The default is 100 ms.

Set IOChannel Digital Output (Serial Pins)

Description Sets the Digital output for serial pins RTS and DTR. This option is only available when the **Line Service** is set to **Signal I/O**.

User Level Admin

Syntax `set iochannel rts|dtr [description <string>]
[failsafe-action none|activate-outut|deactivate-output]
[mode enabled|disabled]`

Options **description**

Provide a description of the channel, making it easier to identify. The channel description can be up to 20 characters.

failsafe-action

When there has been no I/O activity within the specified time (set in the Global Settings) and the **Failsafe Timer** is triggered, you can set the **Failsafe Action** to:

- **None**—The state of the Digital/Relay output remains the same, no change.
- **Activate Output**—Activates the output.
- **Deactivate Output**—Deactivates the output.

mode

Enables the channel, allowing the settings to become active.

Set IOChannel Relay

Description Sets the Relay output channel settings.

User Level Admin

Syntax

```
set iochannel <relay_number> output
[pulse continuous|counted <pulse_count>]
[active-to-inactive-delay <delay>]
[inactive-to-active-delay <delay>]

set iochannel <relay_number>
[active-signal-width <width>] [inactive-signal-width <width>]
[failsafe-action none|activate|deactivate]
```

Options *relay_number*

Specify the Relay channel number, for example, r2.

output

Specify how the channel Digital output will be handled:

- **Manual**—You must manually activate and deactivate the output.
- **Pulse**—When the output is activated, this configuration will cause the output to pulse (go active and inactive) at a configured rate.
- **Inactive-to-Active Delay**—The channel output will remain inactive for the specified time interval after it is manually started.
- **Active-to-Inactive Delay**—The channel output will go inactive after the specified time interval after it is manually started.

The default is **Manual**.

pulse

When the **Output** is **Pulse**, you can have it pulse in a **Continuous** manner or specify a pulse **Count** (each count consists of an active/inactive sequence). The default is **Continuous**.

active-to-inactive-delay

How long to delay an active-to-inactive or inactive-to-active setting after it is manually started. Valid values are 1-9999 x 100 ms. The default is 100 ms.

inactive-to-active-delay

How long to delay an active-to-inactive or inactive-to-active setting after it is manually started. Valid values are 1-9999 x 100 ms. The default is 100 ms.

active-signal-width

How long the channel output will be active during the pulse mode. Valid values are 1-9999 x 100 ms. The default is 100 ms.

inactive-signal-width

How long the channel output will remain inactive during pulse mode. Valid values are 1-9999 x 100 ms. The default is 100 ms.

failsafe-action

When there has been no I/O activity within the specified time (set in the Global Settings) and the **Failsafe Timer** is triggered, you can set the **Failsafe Action** to:

- **None**—The state of the Digital/Relay output remains the same, no change.
- **Activate Output**—Activates the output.
- **Deactivate Output**—Deactivates the output.

Set IOChannel Analog (True Analog)

Description Sets the Analog input channel settings.

User Level Admin

Syntax `set iochannel <analog_channel> type current|voltage
range <range_specifier>`

```
set iochannel <analog_channel> alarm
[level 1|2|3|4|5 [mode on|off] [trigger-type disabled|low|high]
[trigger-level <decimal_value>] [clear-mode auto|manual]
[clear-level <decimal_value>] [snmp on|off] [syslog on|off]]
```

Options `analog_channel`

Specify the Analog channel number, for example, a2 or a4 (this also applies to Temperature models).

type

Select the type of input being measured, either **Current** or **Voltage**. The default is **Current**.

range

Select the range for the measurement type. For current, the range is:

- 0-20 (0-20mA) This is the default.
- 4-20 (04-20mA)

For voltage, the range is:

- 1 (+/-1V)
- 5 (+/-5V)
- 10 (+/-10V) This is the default.
- 150 (+/-150mV)
- 500 (+/-500mV)

alarm

Configures alarm settings when the Analog input trigger is activated.

level

You can specify up to five alarm trigger/clear severity levels. If the **Trigger Type** is **Low**, an alarm is triggered when the input drops below the specified **Trigger** value; other severity level trigger values must decrease in value with each subsequent level. If the **Trigger Type** is **High**, an alarm is triggered when the input is higher than the specified **Trigger** value; other severity level trigger values must increase in value with each subsequent level. To clear an alarm, the input must drop below the specified value when **Trigger Type** is **High** or go above the specified value when **Trigger Type** is **Low**.

mode

Enables/disables an alarm level. The default is off.

trigger-type

If the **Trigger Type** is **Low**, an alarm is triggered when the input drops below the specified **Trigger** value; other severity level trigger values must decrease in value with each subsequent level. If the **Trigger Type** is **High**, an alarm is triggered when the input is higher than the specified **Trigger** value; other severity level trigger values must increase in value with each subsequent level.

trigger-level

Specify the value that will trigger an alarm, the measurement is based on the **Type** and **Range** that you specify. This value must not fall within the scope of the value used to clear an alarm.

clear-mode

Specifies whether an activated alarm must be **Manually** cleared, or can be cleared when the input drops below the specified value (when **Trigger Type** is **High**) or goes above the specified value (when **Trigger Type** is **Low**).

clear-level

Specify that value that will clear an alarm, the measurement is based on the **Type** and **Range** that you specify. This value must not fall within the scope of the value used to trigger an alarm.

snmp

Sends an SNMP trap when an alarm is triggered or cleared. The trap consists of the severity level and whether the alarm was triggered or cleared.

syslog

Sends a message to syslog when an alarm is triggered or cleared. The syslog entry includes the severity level and the value that caused the alarm to trigger or clear. The syslog message is associated with **Level Critical**.

Set IOChannel Analog (Temperature)

Description Sets the Analog input channel settings for Temperature models.

User Level Admin

Syntax **set iochannel** *<analog_channel>* **type** **rtd|thermocouple**
range *<range_specifier>*

```
set iochannel <analog_channel> alarm
[level 1|2|3|4|5 [mode on|off] [trigger-type disabled|low|high]
[trigger-level <decimal_value>] [clear-mode auto|manual]
[clear-level <decimal_value>] [snmp on|off] [syslog on|off]]
```

Options *analog_channel*

Specify the Analog channel number, for example, a2 or a4 (this also applies to Temperature models).

type

Specify the type of sensor you are using to measure temperature, either RTD or thermocouple. The default is RTD.

range

Specify the temperature range that you want to measure. For RTD, the range is:

- 1 (Pt100 a=385 -50 to 150C) This is the default.
- 2 (Pt100 a=385 0 to 100C)
- 3 (Pt100 a=385 0 to 200C)
- 4 (Pt100 a=385 0 to 400C)
- 5 (Pt100 a=385 -200 to 200C)
- 6 (Pt100 a=392 -50 to 150C)
- 7 (Pt100 a=392 0 to 100C)
- 8 (Pt100 a=392 0 to 200C)
- 9 (Pt100 a=392 0 to 400C)
- 10 (Pt100 a=392 -200 to 200C)
- 11 (Pt1000 a=385 -40 to 160C)
- 12 (NiFe604 a=518 -80 to 100C)
- 13 (NiFe604 a=518 0 to 100C)

For thermocouple, the range is:

- b (B 500 to 1800C)
- e (E 0 to 1000C)
- j (J 0 to 760C) This is the default.
- k (K 0 to 1370C)
- r (R 500 to 1750C)
- s (S 500 to 1750C)
- t (T -100 to 400C).

alarm

Configures alarm settings when the Analog input trigger is activated.

level

You can specify up to five alarm trigger/clear severity levels. If the **Trigger Type** is **Low**, an alarm is triggered when the input drops below the specified **Trigger** value; other severity level trigger values must decrease in value with each subsequent level. If the **Trigger Type** is **High**, an alarm is triggered when the input is higher than the specified **Trigger** value; other severity level trigger values must increase in value with each subsequent level. To clear an alarm, the input must drop below the specified value when **Trigger Type** is **High** or go above the specified value when **Trigger Type** is **Low**.

mode

Enables/disables an alarm level. The default is off.

trigger-type

If the **Trigger Type** is **Low**, an alarm is triggered when the input drops below the specified **Trigger** value; other severity level trigger values must decrease in value with each subsequent level. If the **Trigger Type** is **High**, an alarm is triggered when the input is higher than the specified **Trigger** value; other severity level trigger values must increase in value with each subsequent level.

trigger-level

Specify the value that will trigger an alarm, the measurement is based on the **Type** and **Range** that you specify. This value must not fall within the scope of the value used to clear an alarm.

clear-mode

Specifies whether an activated alarm must be **Manually** cleared, or can be cleared when the input drops below the specified value (when **Trigger Type** is **High**) or goes above the specified value (when **Trigger Type** is **Low**).

clear-level

Specify that value that will clear an alarm, the measurement is based on the **Type** and **Range** that you specify. This value must not fall within the scope of the value used to trigger an alarm.

snmp

Sends an SNMP trap when an alarm is triggered or cleared. The trap consists of the severity level and whether the alarm was triggered or cleared.

syslog

Sends a message to syslog when an alarm is triggered or cleared. The syslog entry includes the severity level and the value that caused the alarm to trigger or clear. The syslog message is associated with **Level Critical**.

Kill IOChannel

Description Kills the I/O channel.

User Level Admin

Syntax `kill iochannel <i/o_channel>`

Options `kill iochannel line <number> rts|cts|dtr|dsr|dcd`

i/o_channel

Specify the channel number, for example, d2 or a4. Temperature models use Analog input, so the channel numbers are a1-a4.

rts|cts|dtr|dsr|dcd

Specify the Digital output pins (RTS or DTR) or Digital input pins (CTS, DSR, or DCD).

Show IO

Description Shows global I/O information (for example, UDP, TruePort, Modbus). Temperature input is Analog.

User Level Admin

Syntax `show iochannel <i/o_channel>`

Options `show iochannel rts|cts|dtr|dsr|dcd`

i/o_channel

Specify the channel number, for example, d2 or a4. Temperature models use Analog input, so the channel numbers are a1-a4.

rts|cts|dtr|dsr|dcd

Specify the Digital output pins (RTS or DTR) or Digital input pins (CTS, DSR, or DCD).

Show IOChannel

Description Shows I/O channel information. Temperature input is Analog.

User Level Admin

Syntax `show iochannel <i/o_channel>`

Options `show iochannel line <number> rts|cts|dtr|dsr|dcd`

i/o_channel

Specify the channel number, for example, d2 or a4. Temperature models use Analog input, so the channel numbers are a1-a4.

rts|cts|dtr|dsr|dcd

Specify the Digital output pins (RTS or DTR) or Digital input pins (CTS, DSR, or DCD).

I/O Channel Control Commands

The I/O commands in this section are used to manually manage the I/O channels.

Digital Output

Description Manages the Digital output channel status. Not all models have four digital channels, most have just two.

User Level Admin

Syntax `iochannel d1|d2|d3|d4|cts|dsr|dcd clear alarm|input-latch`

Options **alarm**

Clears the alarm. Note that if the condition that tripped the alarm still exists, the alarm will not look like it's cleared, but will reflect the appropriate alarm level severity. Alarm Level 0 means that the alarm has not been triggered.

latch-input

Clears the latch value.

Digital Input

Description Manages the Digital input channel status.

User Level Admin

Syntax `iochannel d1|d2|d3|d4|rts|dtr output activate|deactivate`

Option **output**

Manually activates/deactivates the I/O channel.

Relay

Description Manages the Relay output channel status.

User Level Admin

Syntax `iochannel r1|r2 output activate|deactivate`

Option **output**

Manually activates/deactivates the I/O channel.

Analog Input

Description Manages the Analog input channel status.

User Level Admin

Syntax `iochannel a1|a2|a3|a4 clear alarm|min|max|input-latch`

Options **alarm**

Clears the alarm. Note that if the condition that tripped the alarm still exists, the alarm will not look like it's cleared, but will reflect the appropriate alarm level severity. Alarm Level 0 means that the alarm has not been triggered.

min

Clears the minimum value.

max

Clears the maximum value.

latch-input

Clears the latch value.

Calibrating Analog Input (Analog/Temperature)

Calibrate Analog

Description Calibrates the Analog input channel. When this command is issued, a script will automatically start, requesting that the minimum and maximum calibration values be applied to the requested Analog channel. See [Calibrating Analog Input on page 89](#) or [Calibrating Temperature Input on page 89](#) for more information.

User Level Admin

Syntax `iochannel a1|a2|a3|a4 calibrate`

Reset Calibration

Description Resets the calibration to factory defaults.

User Level Admin

Syntax `reset io calibration`



Troubleshooting

Introduction

This chapter provides information that can help resolve problems with the Device Server.

Hardware Problems

If the Device Server Power/Ready LED is red and stays red for over 10 seconds, you have a hardware problem that might require factory service. First, try the following:

- If the Device Server is not in Console mode, do the following:
 - a. Set up a direct connection to the Device Server; see [Using a Direct Connection](#) on page 40 for information on this type of connection.
 - b. Power the Device Server off.
 - c. Switch the Console dip switch to On.
 - d. Power the Device Server on.

If there is a problem with the Device Server firmware, you will need to reload the firmware, which can be found either on the CD-ROM that came with the Device Server or on the Perle website, www.perle.com/downloads/serial.shtml.

- If the Device Server is already in Console mode and the Power LED stays red, you need to make arrangements to return the Device Server.

If you purchased the Device Server less than 30 days before this problem appears, contact your distributor; otherwise, see the Perle web site (www.Perle.com) for factory service information.

Note: no factory service can be done on a Device Server that has not been registered.

Communication Issues

General communication checks and practices are as follows:

- Are your cables connected and correctly configured? If you are using EIA-232, see [EIA-232 Cabling Diagrams](#) on page 46 to verify that your cables are correctly configured.
- Ping your host? If you can ping but packet loss is reported, ping another host/device on the same network. This will tell you whether the problem is specific to the host/device or general to the network.
- After entering or changing IP information for your Device Server, *reboot* the Device Server (does not apply when using BOOTP or DHCP). Once the Device Server has rebooted, other network devices should be able to communicate with it (ping, telnet, etc.). Also, protocols such as ARP and proxy-ARP will work properly.
- Use the **show routes** command (command line only) or view the **Routes** statistics. Is there a route to the host?
- If the WebManager or DeviceManager cannot communicate with the Device Server, verify that the **Server Services HTTP** and/or **HTTPS** are enabled for WebManager and **DeviceManagerD** is enabled for DeviceManager.

DeviceManager Problems

Error Message: **16 bit Windows Subsystem - C:\WINDOWS\SYSTEM32\AUTOEXEC.NT. The system file is not suitable for running MS-DOS and Microsoft Windows applications. Choose 'Close' to terminate the application.**

The error message can be misleading, because it is displayed even if the **AUTOEXEC.NT** file is actually missing.

To verify whether you have the file, type **%windir%/system32/** in the address bar of an Explorer window. If there is no **AUTOEXEC.NT** file proceed as follows:

1. Browse to **%windir%/repair/** (usually **C:\WINDOWS\repair**).
2. Right-click and Copy the **AUTOEXEC.NT** file.
3. Browse to **%windir%/system32/** (usually **C:\WINDOWS\System32**).
4. Right-click inside the window and Paste the file.

The error condition described here may also be the result of corruption of the **AUTOEXEC.NT** file, in which case the above procedure may be helpful to restore a valid file.

If the above procedure does not fix the DeviceManager installation problem, see <http://support.microsoft.com/?kbid=324767> for the official Microsoft explanation.

Host Problems

Cannot access a host on a local network, verify:

- The network address is correct.
- The subnet mask is set correctly and reflects the network configuration.
- The broadcast address is set correctly and reflects the network configuration.

Cannot access a host on a remote network:

- Use the **show route** command to verify that there is a route to the remote host. If no gateway is specified, verify that a default gateway is specified. Ping the default gateway to check if it is working.
- Consider the situation beyond the gateway; for example, are intermediate gateways and the remote host available? Also, check the messages returned by the **ping** command; for example, that a particular host or gateway is unreachable.

Gateways added into the gateway table are ignored by the Device Server:

- Have you used BOOTP and entered a single static gateway in the bootptab file entry? If yes, the other gateways will be ignored.

Access to host lost after a few minutes.

- If the route to this host goes through routers, make sure those routers are all sending RIP packets across the networks.

Login Problems

You have lost or don't know your password (as Admin user).

- You must reset the Device Server to its factory default settings using the **Reset** switch on the rear panel. There is no procedure to access the Device Server without a password.

Problems with Terminals

The following section concerns problems with the appearance of data on your terminal screen.

The Device Server logs me out after a few minutes:

- Check the **Idle Timer** value set for the user. The default setting for the **Idle Timer** for all users is 0 seconds (does not timeout).

Corrupt data.

- Check your line settings (baud rate, stop bits, etc.)

Missing data.

- Verify that the same type of flow control is set in both your terminal and on the Device Server's port.

Error message not permitted on a dumb terminal after typing the CLI command screen.

- Set your **Line** to **Termttype** VT100, ANSI or WYSE60 (or other form of terminal emulation, if you have downloaded one). The default line type in the Device Server is **Dumb**, which does not support the graphics characters necessary to view the text-based menus.

Screen corruption when using the text-based menu system.

- Verify that the terminal setup in the Device Server matches your terminal.
- Verify that entries in the term file match your terminal setup.
- If using a PC/computer, verify that the type of terminal emulation selected in your application matches those supported by the Device Server.

When using the function keys on your keyboard, nothing happens or your sessions keep swapping.

- Change your **Hotkey Prefix** character. The function keys on the keyboards of some terminals (like WYSE60) send character sequences which begin with **^a**; unfortunately, **^a** is also the default **Hotkey Prefix**, which you use to switch between sessions. A valid alternative would be **^b** (hex=02). If you are the system administrator, you can change any user's **Hotkey Prefix** character.

When using a downloaded terminal definition, you are having problems using arrow keys.

- Use Ctrl-K, Ctrl-J, Ctrl-H and Ctrl-L for up, down, left and right respectively.

When switching from a session back to the text menus, both screen images are superimposed.

- Press **^r** to redraw the screen.

INIT: Error in terminal file <filename>

- This error indicates that you have exceeded the 80 character limit for one or more of the terminal capabilities defined in the reported file.

INIT: Error on line *n* in terminal file <filename>

- You have omitted the = sign from the reported line.

Unknown IP Address

You have a Device Server already configured and you do know your password, but have lost, misconfigured, or don't know the IP address of the Device Server, and you cannot obtain a login.

- If the Device Server resides within the local network segment, you can use DeviceManager to find the Device Server.
- You can connect directly to the serial port of the Device Server, as explained in [Using a Direct Connection on page 40](#).

DHCP/BOOTP Problems

Messages: host name too long or filename too long.

- The Device Server can only accept host names of 14 characters or file names of 64 characters, so verify that you are not attempting to pass a string that is longer than those maximums.

DHCP or BOOTP have been set up to configure my Device Server, but does not seem to have done anything.

- Check that the server DHCP/BOOTP service is set to on, if not set it to on and reboot.
- Check that your BOOTP server is configured for your Device Server or that your DHCP server has an active lease pool (scope) with at least 1 free IP address.

You observe TFTP errors when the Device Server boots, for example:

TFTP: File not found : filename

TFTP: Timed out

This has a number of causes, including:

- The file names you specified to DHCP/BOOTP do not exist or are in the wrong place.
- The server for any of the downloadable files in your bootfile has no TFTP server running.
- Verify that lease data in your DHCP server manager is correct.
- Reset or restart the DHCP server.

Language Problems

In a customised language, the text strings appear in the wrong place in the Menu, CLI, or WebManager.

- Check the original ASCII text file you used to translate to your customised language. The sequence of the line much match exactly (be aware that comments don't affect line sequence, but can affect the actual line that the strings appear on). So, if you strip out all comments, if the original file says line 1000 should be string **none**, then line 1000 (stripped of comments) should be the translated version of **none**.

Long Reboot Cycle

Rebooting the Device Server takes a long time.

If you are not using DHCP/BOOTP, disable this within the Server Services; otherwise, the Device Server waits to timeout for a request to DHCP/BOOTP.

I/O Models

An I/O Digital or Relay controlled motor is starting/stopping

- Digital and Relay channels have automatically resetting fuses, meaning that if the circuit gets overloaded and the fuse blows, it will automatically reset when the circuit cools down.

An A4R2 model is starting/stopping

- The A4R2 model can run at 55 degrees Celsius ambient temperature when the input voltage is 22VDC or below. If the input voltage exceeds 22VDC, the maximum ambient temperature will drop into the range of 45-50 degrees Celsius to run successfully.



Utilities

Introduction

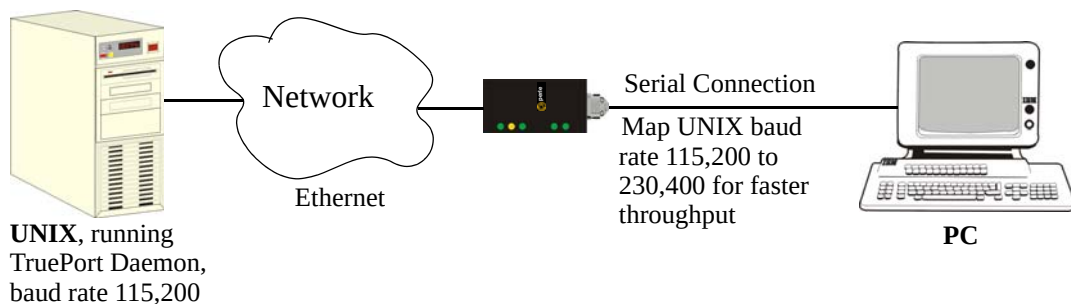
This chapter provides information on the TruePort utility.

TruePort

TruePort is a com port redirector utility for the Device Server. It can be run in two modes:

- **TruePort Full mode**—This mode allows complete device control and operates exactly like a directly connected serial port. It provides a complete COM port interface between the attached serial device and the network.
- **TruePort Lite mode**—This mode provides a simple raw data interface between the device and the network. Although the port will still operate as a COM port, control signals are ignored. In this mode, the serial communications parameters must be configured on the Device Server.

You use TruePort when you want to connect extra terminals to a server using a Device Server rather than a multi-port serial card. TruePort is especially useful when you want to improve security, for example, to see which user is logged onto which terminal. When run on UNIX, TruePort allows you to print directly from a terminal to an attached printer (transparent printing). You can also remap the slow baud rate of your UNIX server to a faster baud rate, as shown below.



Currently, TruePort is supported on Linux, Windows, SCO, Solaris, and others. For a complete list of supported operating systems, see the Perle website.

For more information, see the *TruePort User Guide* or the *TruePort Installation and Configuration Guide for Windows NT* on the CD-ROM.

API I/O Access Over TruePort

You can access Device Server I/O data through TruePort using the Perle API. The API uses the command/response format. See the `ioapiottp.c` sample program, found on the product CD-ROM, for an example implementation.

API Request Format

All data in the Request must be sent as a single write to the COM port. The API command takes the following format:

Number of Bytes	Value
1	Function Code (in hex): ● 01—Get read/write boolean register ● 03—Get read/write register ● 04—Get read only register ● 15—Set read/write boolean register (0x0F) ● 16—Set read/write register (0x10)
2	Starting register number (see A4/T4 Registers on page 79, A4D2/A4R2 Registers on page 80, or D4/D2R2 Registers on page 81 for this value).
2	Number of registers to act on starting from the register defined in byte 2 above.
n	Data for write. Some values must be read/written as a unit consisting of 2 consecutive registers. If the request is to write, the data to write follows the number of registers. If accessing registers consisting of 2 bytes or 4 bytes, the data is in big endian (network order) format.

API Response Format

The API command takes the following response format:

Number of Bytes	Value
1	Function code of request if no error. Most significant bit will be set if an error occurred.
1	Length of data in response if no error occurred. If an error occurred, the byte will contain the error code (see the Error Codes table below).
n	Data response for request (the number of bytes is dependent on the number and type of registers requested). If the request returns 2 or 4 byte values, it will be in big endian (network order) format. If the request returns boolean values, the least significant bit (bit 0) represents the first value requested and bits 1 to 7 represents subsequent boolean values. If more than 8 boolean registers are requested, they are returned in successive bytes.

Error Codes

Code	Name	Description
01	Illegal Function	The function code received in the query is not an allowable action for the server (or slave).
02	Illegal Data Address	The data address received in the query is not an allowable address for the server (or slave).
03	Illegal Data Value	A value contained in the query data field is not an allowable value for server (or slave).
04	Slave Device Failure	An unrecoverable error occurred while the server (or slave) was attempting to perform the requested action.



Accessories

Introduction

This chapter provides information about peripheral Device Server options that can be ordered separately from the product. Contact your sales representative to find out how to order the products listed in this appendix.

Starter Kit (Adapters/Cable)

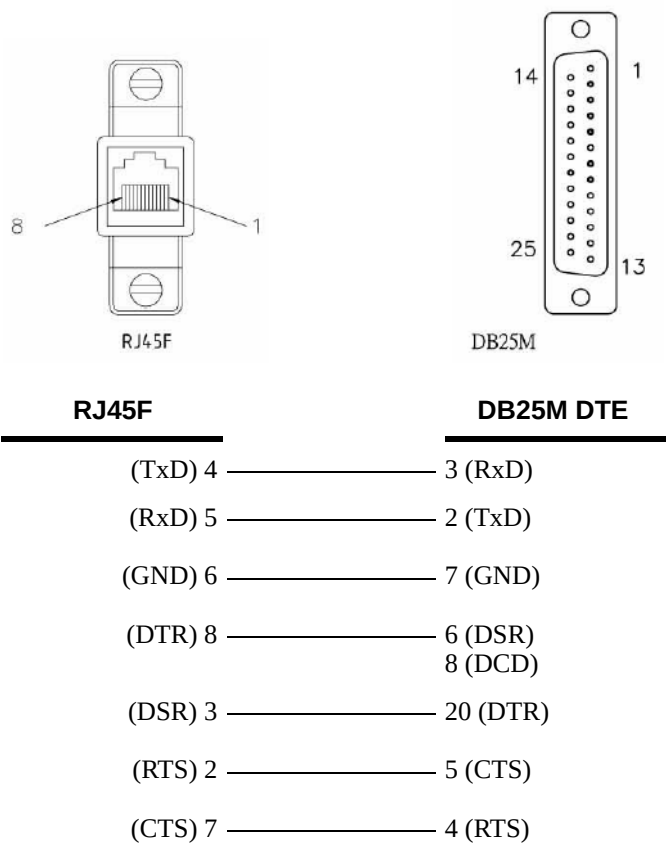
The IOLAN Starter Kit includes the following:

- *RJ45F to DB25M DTE Crossover Adapter*
- *RJ45F to DB25M DCE Modem Adapter*
- *RJ45F to DB25F DTE Crossover Adapter*
- *RJ45F to DB9M DTE Crossover Adapter*
- *RJ45F to DB9F DTE Crossover Adapter*
- *Sun/Cisco RJ45M Connector Cable for Rack Mount Models*

The adapters/cable can be purchased as a kit or individually.

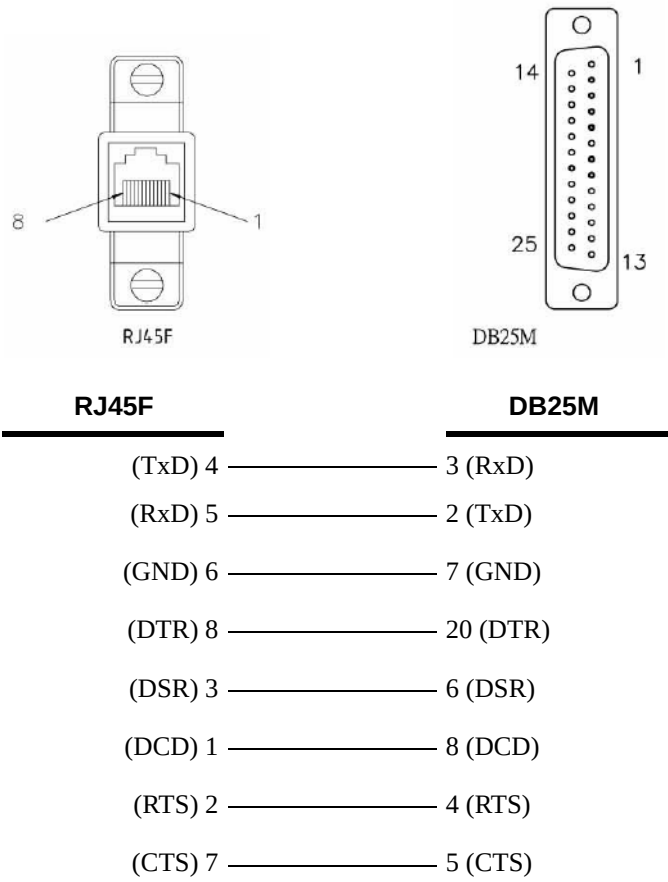
RJ45F to DB25M DTE Crossover Adapter

The following diagram shows the IOLAN Device Server RJ45F-->DB25M DTE crossover adapter pinouts. This is model number DBA0011.



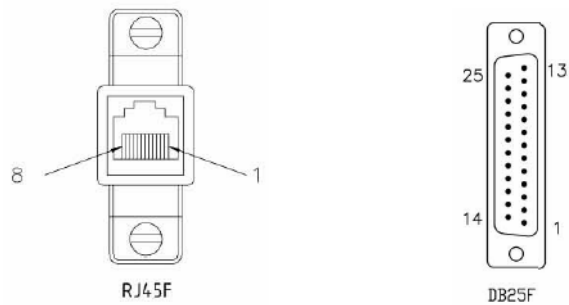
RJ45F to DB25M DCE Modem Adapter

The following diagram shows the IOLAN Device Server RJ45F→DB25M DCE modem adapter pinouts. This is model number DBA0013.



RJ45F to DB25F DTE Crossover Adapter

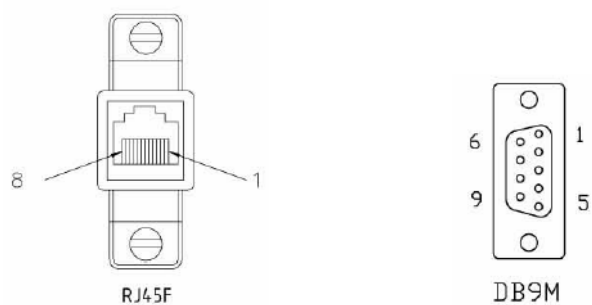
The following diagram shows the IOLAN Device Server RJ45→DB25F DTE crossover adapter pinouts. This is model number DBA0010.



RJ45F	DB25F
(TxD) 4	3 (RxD)
(RxD) 5	2 (TxD)
(GND) 6	7 (GND)
(DTR) 8	6 (DSR) 8 (DCD)
(DSR) 3	20 (DTR)
(RTS) 2	5 (CTS)
(CTS) 7	4 (RTS)

RJ45F to DB9M DTE Crossover Adapter

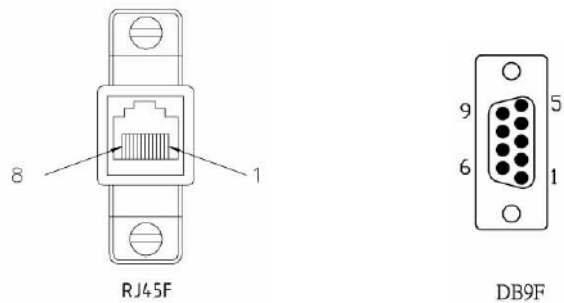
The following diagram shows the IOLAN Device Server RJ45→DB9M crossover adapter pinouts. This is model number DBA0021.



RJ45F	DB9M
(Tx) 4	2 (Rx)
(Rx) 5	3 (Tx)
(GND) 6	5 (GND)
(DTR) 8	1 (DCD)
	6 (DSR)
(DSR) 3	4 (DTR)
(RTS) 2	8 (CTS)
(CTS) 7	7 (RTS)

RJ45F to DB9F DTE Crossover Adapter

The following diagram shows the IOLAN Device Server RJ45F→DB9F crossover adapter pinouts. This is model number DBA0020.



RJ45F	DB9F
(TxD) 4	2 (RxD)
(RxD) 5	3 (TxD)
(GND) 6	5 (GND)
(DTR) 8	1 (DCD)
	6 (DSR)
(DSR) 3	4 (DTR)
(RTS) 2	8 (CTS)
(CTS) 7	7 (RTS)

Sun/Cisco RJ45M Connector Cable for Rack Mount Models

This is a 3 meter RJ45M→RJ45M 8-wire Sun/Cisco modular cable. The following diagram shows how the IOLAN DS RJ45M cable is configured when connecting to the supplied Sun/Cisco RJ45 cable. This model number is CAB0030.

IOLAN DS RJ45M	Sun/Cisco RJ45M
(RTS) 2	8 (CTS)
(DSR) 3	2 (DTR)
(TxD) 4	6 (RxD)
(RxD) 5	3 (TxD)
(GND) 6	4 (GND)
(CTS) 7	1 (RTS)
(DTR) 8	7 (DSR)



Glossary

This chapter provides definitions for Device Server terms.

BOOTP (BOOTstrap Protocol)	An Internet protocol that enables a diskless workstation to discover its own IP address, the IP address of a BOOTP server on the network, and a file to be loaded into memory to boot the machine. This enables the workstation to boot without requiring a hard or floppy disk drive.
Community (SNMP)	An SNMP community is the group that devices and management stations running SNMP belong to. It helps define where information is sent.
DHCP (Dynamic Host Configuration Protocol)	A TCP/IP protocol that provides static and dynamic address allocation and management.
Direct Connection	Connections that bypass the Device Server enabling the user to log straight into a specific host. A direct connection is recommended where a user logging in to the Device Server is not required.
Ethernet	A high-speed (10Mbps,100Mbps) cable technology that connects devices to a LAN, using one or more sets of communication protocols.
Local Authentication	Uses the user ID and password stored within the Device Server User database.
Modem Initialization String	A series of commands sent to the modem by a communications program at start up. These commands tell a modem how to set itself up in order to communicate easily with another modem.
MOTD	Message of the day. This is defined by a file whose contents display when users log into the Device Server.
Multicast	The broadcasting of messages to a specified group of workstations on a LAN, WAN, or internet.
NAK (Negative Acknowledgment)	A communication control character sent by the receiving destination indicating that the last message was not received correctly.
Reverse Connection	Connections that originate from a host that go directly to a serial device through the Device Server.
Silent Connection	Silent connections are the same as direct connections except that they are permanently established. The host login prompt is displayed on the screen. Logging out redispays this prompt. Silent connections, unlike direct connections, however, make permanent use of pseudo tty resources and therefore consume host resources even when not in use.
SNMP (Simple Network Management Protocol)	A protocol for managing network devices.
Subnet/Prefix Bits	Identifies the device's IP address, which portion constitutes the network address and which portion constitutes the host address.



Index

A

admin

- default password [39](#)
- level [67](#)
- lost password [73](#)

alarms, I/O [75](#)

analog

- calibrating [89](#)
- input [86](#)

API

- I/O commands [206](#)
- TruePort [82](#)

ARP-Ping, setting an IP address [41](#)

B

bidir

- general [64](#)
- parameters [105](#)

binary configuration file [94](#)

BOOTP

- parameters [74](#)
- setting an IP address [40](#)

bootup files, configuring [134](#)

browsers, supported [25](#)

C

cabling, EIA-232 [46](#)

calibrating

- analog [89](#)
- temperature [89](#)

channels

- analog [86](#)
- digital [83](#)
- relay [87](#)
- temperature [85](#)

CLI

- command shortcuts [144](#)
- syntax [143](#)

client tunnel

- parameters [108](#)

configuration files

- downloading to multiple servers [136](#)
- editing [95](#)
- formats [94](#)

configuring hardware [56](#)

configuring multiple Device Servers [136](#)

connecting to Device Server

- setting IP address [38](#)

connecting to the Device Server

- console mode [29](#)
- serial mode [29](#)

connections

- direct/silent/reverse [64](#)
- dslogin [63](#)

console mode [29](#)

custom app

- creating [74](#)
- SDK [74](#)

D

DB25

- pinouts
 - female [43](#)
 - male [42](#)
- power in pin
 - female connector [43](#)
 - male [42](#)

DB9 male pinouts [44, 45](#)

definitions [215](#)

Device Server models [23](#)

Device Servers, configuring multiple [136](#)

DeviceManager

- overview [51](#)
- setting an IP address [39](#)

DHCP

- parameters [74](#)
- setting an IP address [40](#)

digital

- I/O [83](#)

direct connect

- setting an IP address [40](#)

direct connections [64](#)

dslogin [63](#)

E

easy port access menu [67](#)
ethernet configuration [56](#)

F

factory defaults, resetting to [73](#)
files, downloading [73](#)

G

gateway parameters [132](#)

H

hardware configuration [56](#)
host parameters [130](#)

I

I/O

- alarms [75](#)
- analog [86](#)
- digital [83](#)
- Modbus [77](#)
- relay [87](#)
- temperature [85](#)
- UDP [75](#)

I/O SNMP traps [88](#)
IPv6, setting an IP address [41](#)

J

jumpers

- line termination [31](#)
- power out [31](#)
- setting [31](#)

L

language

- translating [70](#)
- upgrading firmware [71](#)

LED guide [29](#)
levels, user [67](#)
line access parameters [129](#)
line parameters [99](#)
line termination, setting jumper [31](#)

M

Menu

- conventions [53](#)
- using [52](#)

menu level [67](#)
MIB [54](#)

Modbus

- configuration overview [57](#)
- example scenario [60](#)
- gateway settings [58](#)
- I/O access [77](#)
- line settings [59](#)
- TruePort [82](#)

modbus master

- parameters [110](#)

modbus slave

- parameters [109](#)

mode

- console [29](#)
- serial [29](#)

models, Device Server

[23](#)

modem parameters

[116](#)

MOTD parameters

[134](#)

multisessions

[68](#)

N

normal level [67](#)

O

online help, using [20](#)

P

packet forwarding

- parameters [114](#)

parameters

- bidir [105](#)
- BOOTP/DHCP [74](#)
- bootup files [134](#)
- client tunnel [108](#)
- gateways [132](#)
- hardware [98](#)
- hosts [130](#)
- line [99](#)
- line access [129](#)
- modbus master [110](#)
- modbus slave [109](#)
- modems [116](#)
- MOTD [134](#)
- packet forwarding [114](#)
- raw [104](#)
- server [96](#)
- server tunnel [108](#)
- sessions [129](#)
- SNMP [131](#)
- syslog [133](#)
- telnet [104](#)
- TFTP [132](#)
- TruePort [113](#)
- UDP [106](#)
- user [127](#)
- vmodem [107](#)

password
 admin default 39
 lost 73
pin, power in 43
 DB25 male 42
 serial RJ45 44
pinouts
 DB25 female 43
 DB25 male 42
 DB9 male 44, 45
 RJ45 serial 44
power in pin
 DB25 female 43
 DB25 male connector 42
 serial RJ45 44
power out, setting jumper 31
product repair 22

R

raw parameters 104
relay I/O 87
resetting to factory defaults 73
restricted level 67
reverse connections 64
reverse sessions 68
reverse, sessions 68
RJ45
 serial pinouts 44
RJ45 serial power in pin 44

S

SDK, custom application 74
serial configuration 56
serial mode 29
serial tunnel 66
server parameters 96
server tunnel
 parameters 108
services 55
 bidir 64
 line
 signal I/O 64
 TruePort 64
 serial tunnel 66
 UDP 65
 vmodem 64
session parameters 129
sessions 67
setting an IP address
 ARP-Ping 41
 BOOTP/DHCP 40
 DeviceManager 39
 direct connect 40
 IPv6 41
signal I/O
 general 64
silent connections 64

SNMP
 I/O traps 88
 parameters 131
 supported traps 131
 using 54
syslog parameters 133

T

technical support
 contacting 21
 online 21
 product information 21
 product repair 22
 via the internet 21
telnet parameters 104
temperature
 calibrating 89
 input 85
terminal definitions
 creating 71
 downloading 71
text configuration file 94
TFTP 73
TFTP parameters 132
traps, SNMP 131
TruePort
 API 82
 general 64
 Modbus 82
TruePort parameters 113
TruePort utility 205

U

UDP
 configuring 65
 parameters 106
UDP, I/O 75
user levels 67
user parameters 127
user sessions 67
utility
 TruePort 205

V

virtual modem 64
vmodem
 overview 64
 parameters 107

W

WebManager
 overview 52
 using 52

